

Bloque Modular de
INFORMATICA BASICA

Manejo del Sistema Operativo

ANEXOS





Esta obra está bajo una [Licencia Creative Commons
Atribución-NoComercial-CompartirIgual 4.0 Internacional](https://creativecommons.org/licenses/by-nc-sa/4.0/).



Ministerio de Trabajo y Seguridad Social
Servicio Nacional de Aprendizaje
Dirección General
Subdirección de Formación Profesional y Desarrollo Social
Subdirección de Planeación

Bloque Modular de Informática Básica

Módulo Instruccional: Manejo del Sistema Operativo

Anexos

Santafé de Bogotá D.C., abril de 1992

Publicación del Servicio Nacional de Aprendizaje - SENA
Dirección General
Subdirección de Formación Profesional y Desarrollo Social
Subdirección de Planeación

Director General:	Hernando Arango Monedero
Subdirector Nacional de Formación Profesional:	Julio César Del Valle Bayona
Subdirector Nacional de Planeación:	Arturo José García Durán
Coordinación:	División del Sector Comercio y Servicios División de Sistemas e Informática Dirección General
Asesoría Técnico - Pedagógica:	Esperanza Marín Castro Luis Miguel León Cáceres Wilson Moreno León Nora Baena Padilla Nidia María Mendoza Muñoz
Desarrollo de Contenidos y Transcripción:	Eufeminiano Cañón Cortés Instructor de Informática Centro Nacional de la Construcción e Industria de la Madera Regional Bogotá - Cundinamarca
Revisión y Ajuste:	Luis Miguel Redondo Ariza Jorge Elécer Guillermo Romero Carlos Alberto Berrío Torres José Antonio Aguilera Jaramillo Nidia María Mendoza Muñoz
Revisión de Estilo y Edición:	Nidia María Mendoza Muñoz Profesional Grupo de Informática Educativa División de Sistemas e Informática Dirección General
Diseño de Carátula:	Carlos Enrique Pinto Barón Jefe División de Sistemas e Informática Regional Bogotá-Cundinamarca
Impresión:	Grupo de Publicaciones SENA - Dirección General

Primera Edición, abril de 1992

Santafé de Bogotá D.C. - Colombia

Indice

ANEXO No. 1	1
EL TECLADO Y LAS TECLAS DE PROPOSITOS ESPECIALES	
ANEXO No. 2	3
ANATOMIA DEL DOS	
ANEXO No. 3	5
EL PROCESADOR DE ORDENES	
ANEXO No. 4	11
ARCHIVOS OCULTOS	
ANEXO No. 5	13
REGISTRO DE ARRANQUE	

ANEXO No. 6	17
EL SISTEMA OPERATIVO	
ANEXO No. 7	19
SISTEMA OPERATIVO EN DISCO DOS	
ANEXO No. 8	27
EL DISCO FLEXIBLE Y LAS UNIDADES DE DISCO FLEXIBLE	
ANEXO No. 9	29
TIPOS DE UNIDADES DE DISCO	
ANEXO No. 10	31
TIPOS DE DISQUETES	
ANEXO No. 11	33
PREPARACION DE DISCOS NUEVOS	
ANEXO No. 12	35
PROTECCION DE DISCOS	
ANEXO No. 13	37
NOMBRES DE DISPOSITIVOS DOS	
ANEXO No. 14	39
UNIDADES DE DISCO	

ANEXO NO. 15	41
DISCOS FLEXIBLES	
ANEXO No. 16	43
FUNCIONES DOS	
ANEXO NO. 17	45
ARCHIVOS DOS	
ANEXO No. 18	47
INICIACION DEL DOS	
ANEXO No. 19	53
COMANDOS DEL DOS	
ANEXO No. 20	57
ARCHIVOS DEL DOS	
ANEXO No. 21	59
COMANDO DIR	
ANEXO NO. 22	63
CARACTERES "COMODIN" DEL DOS	
ANEXO No. 23	67
COMANDO CHKDSK	

ANEXO No. 24	73
COMANDO COPY	
ANEXO No. 25	75
COMANDO COMP	
ANEXO No. 26	79
COMANDO DISKCOPY	
ANEXO No. 27	81
COMANDO DISKCOMP	
ANEXO No. 28	85
COMANDO FORMAT	
ANEXO No. 29	87
COMANDO SYS	
ANEXO No. 30	89
COMANDO TYPE	
ANEXO No. 31	93
USO DE LOS NOMBRES DE DISPOSITIVOS DEL DOS	
ANEXO No. 32	99
COMANDO PRINT	

ANEXO No. 33	101
REDIRECCION DE LA ENTRADA Y DE LA SALIDA	
ANEXO No. 34	107
COMANDO RENAME	
ANEXO No. 35	109
COMANDO DEL	
ANEXO No. 36	111
GESTION DE ARCHIVOS DOS CON SUBDIRECTORIOS	
ANEXO No. 37	119
COMANDO MKDIR	
ANEXO No. 38	121
COMANDO CHDIR	
ANEXO No. 39	125
COMANDO RMDIR	
ANEXO No. 40	127
COMANDO TREE	
ANEXO No. 41	129
COMANDO PATH	

ANEXO No. 42 133
 PROCESAMIENTO DE LOTES
 CREACION DE ARCHIVOS PARA PROCESO DE LOTES

ANEXO No. 43 153
 ADECUACION DEL DOS

ANEXO No. 1

EL TECLADO Y LAS TECLAS DE PROPOSITOS ESPECIALES

La figura muestra una plantilla del teclado IBM PC. Hay varias teclas de propósitos especiales que van a simplificar el uso de su computador. En primer lugar, las teclas que están marcadas como **F1, F2...** hasta **F10**, se denominan *teclas de función*. Como se verá en anexos posteriores, el DOS permite usar estas teclas para simplificar la entrada a través del teclado. Además, la mayoría de los programas de aplicación, tales como los procesadores de texto, utilizan estas teclas para propósitos especiales.

La siguiente tecla importante es la tecla **RETROCESO** (BACKSPACE). La tecla RETROCESO permite borrar la letra que precede al cursor. Por ejemplo, supongamos que su pantalla contiene

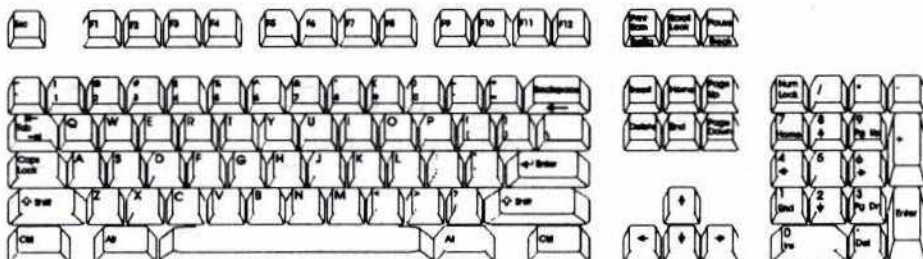
```
A>DISKCOPY
```

Al pulsar una vez la tecla RETROCESO obtendremos:

```
A>DISKCO_
```

Si pulsamos la tecla otra vez, tendremos:

```
A>DISKO_
```

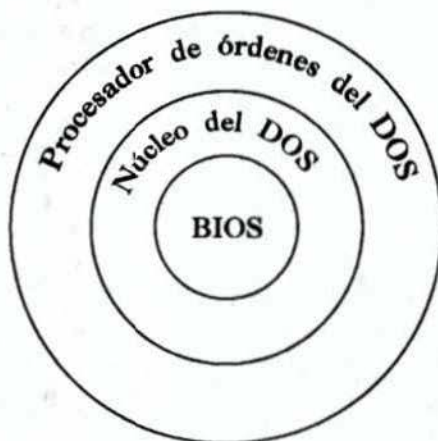
Posición de las teclas de función

NOTA: Como se verá, el DOS utiliza la tecla **ENTER** (INTRO, RETURN o ↵) para finalizar cada línea de comando que se escribe en el teclado. Por ejemplo, si se deseara invocar la orden **DIR** del DOS, habría que digitar **DIR** y después se pulsaría la tecla **ENTER**.

ANEXO No. 2

ANATOMIA DEL DOS

Este anexo y los tres siguientes presentan una descripción más técnica del DOS y de su funcionamiento interno. Si acaba de empezar con el DOS, no pase demasiado tiempo intentando comprender perfectamente la información que se presenta. Sólo es importante que tenga una comprensión básica del procesador de órdenes del DOS, de los procedimientos de arranque del DOS y de las interrupciones del DOS.



Las tres partes funcionales del DOS

La mejor manera de comprender el DOS es examinarlo en términos de sus funciones. Considere el diagrama que se muestra en la figura. El DOS está dividido en tres secciones funcionales. El procesador de órdenes es la parte del DOS que invoca a cada una de las órdenes que se da. El núcleo del DOS proporciona el programa de interfaz con el DOS. El BIOS del DOS proporciona

los servicios de entrada/salida que utiliza el DOS. La mayor parte de los usuarios no tiene que preocuparse nunca por el núcleo del DOS ni por el BIOS. Hay muchos usuarios, sin embargo, a los que les resulta beneficioso comprender el procesador de órdenes del DOS.

ANEXO No. 3

EL PROCESADOR DE ORDENES

El procesador de órdenes del DOS es el programa responsable de llevar a cabo las órdenes que se dan. Por defecto, el DOS utiliza el archivo *COMMAND.COM* como procesador de órdenes. Es posible, sin embargo, especificar un programa distinto como procesador de órdenes, mediante la entrada *COMMSPEC=* del archivo *CONFIG.SYS*

COMMSPEC=ALGUNARC.COM

La mayoría de los usuarios del DOS no necesitará nunca un procesador de órdenes alternativo.

El procesador de órdenes del DOS consta de tres partes: una parte de arranque, la sección residente y la sección transitoria, como se muestra en la figura.



Memoria

Partes del procesador de órdenes del DOS

La parte de arranque del procesador de órdenes del DOS existe únicamente para ejecutar el procedimiento **AUTOEXEC.BAT** si ese procedimiento está presente en el directorio raíz. Cuando termina el procedimiento **AUTOEXEC.BAT**, la parte de arranque del procesador de órdenes concluye y el espacio que consumía en memoria se libera para ser utilizado por otros programas.

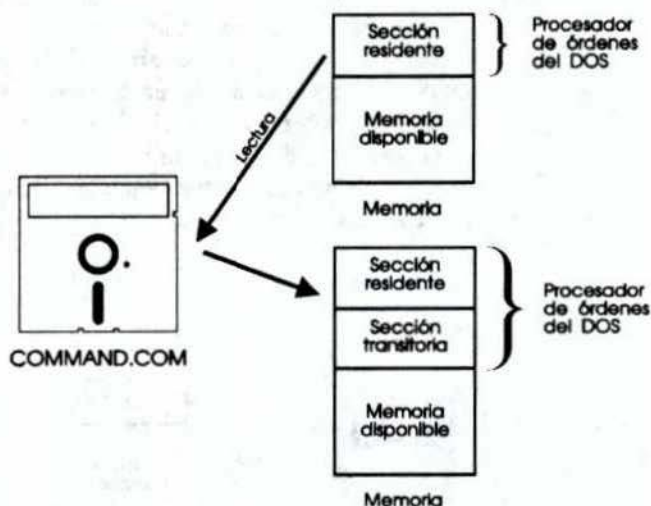
El DOS mantiene la parte residente del procesador de órdenes en memoria en todo momento. Esta porción desempeña todas las funciones esenciales a las que el DOS tiene que ser capaz de dar respuesta inmediata, tales como la carga de la parte transitoria del procesador de órdenes, la conclusión de la ejecución de un programa o la terminación de los programas por procesamiento de **CTRL-BREAK**. Además, la parte residente del procesador de órdenes lleva a cabo la gestión de errores críticos del DOS.

Este procedimiento es el responsable del mensaje

ERROR: Unidad no preparada, leyendo la unidad A
¿Cancelar, reintentar o ignorar?

El mensaje aparece cuando el DOS ya no puede continuar sin intervención del usuario.

La tercera parte del procesador de órdenes del DOS es la sección transitoria. Como se verá, hay muchas aplicaciones del DOS que escriben encima de esta sección del procesador de órdenes, que está en memoria. Cuando esto ocurre, la sección residente del procesador vuelve a cargar la parte transitoria a partir del disco, como se muestra en la siguiente figura:



La parte residente del procesador de órdenes volviendo a cargar la parte transitoria

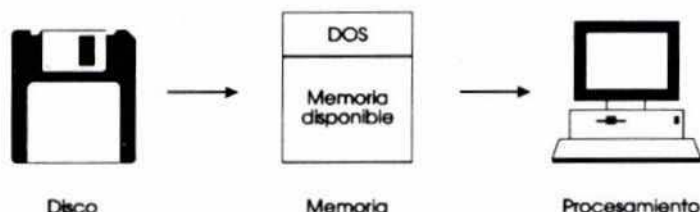
Si la parte residente del DOS no puede encontrar el procesador de órdenes (que normalmente es **COMMAND.COM**), entonces se verá el mensaje:

Inserte disquete con **COMMAND.COM** en unidad A
pulse cualquier tecla para continuar

Limítese a colocar un disco que contenga el procesador de órdenes en la unidad de disco especificada y pulse cualquier tecla.

La parte transitoria del procesador de órdenes recibe órdenes del usuario interactivo o de archivos por lotes del DOS. Esta sección tiene la responsabilidad de ejecutar todas las órdenes del DOS.

Todas las órdenes del DOS se pueden clasificar, bien como *internas* o bien como *externas*. Las órdenes *internas* del DOS están residentes en memoria dentro del procesador de órdenes del DOS. Cuando se invoca a una de estas órdenes, el DOS se limita a localizar la orden en memoria y a ejecutarla. Las órdenes *externas*, por otra parte, residen en disco. Cada vez que se invoca a una orden externa, el DOS tienen que cargar la orden del disco en la memoria, como se ilustra en la figura:



Ordenes externas del DOS

La parte transitoria del procesador de órdenes del DOS contiene todas las órdenes internas del DOS. Estas son:

BREAK	PATH
CHDIR	PROMPT
CLS	REN
COPY	RMDIR
CTTY	SET
DATE	TIME
DEL	TYPE
DIR	VER
ERASE	VERIFY
MKDIR	VOL

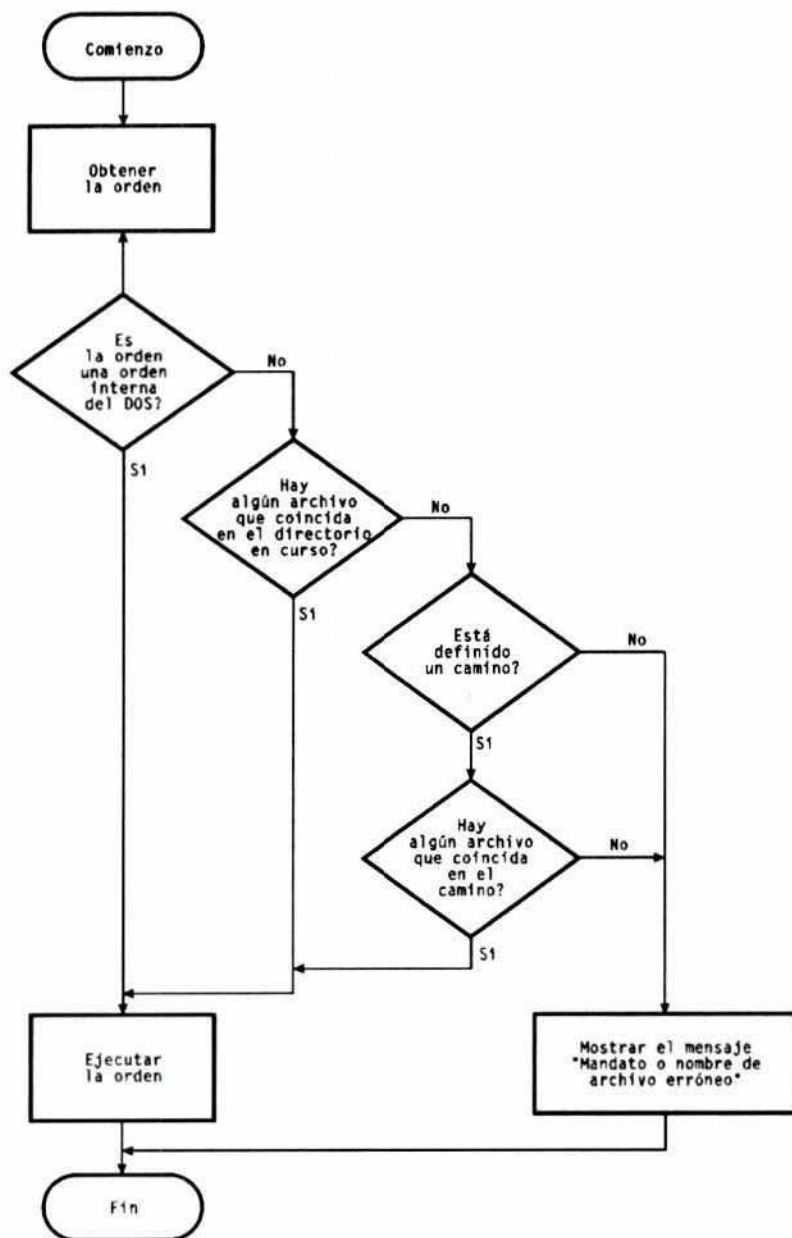
El DOS tiene una lista con estas órdenes. Utilizando esta lista, puede determinar rápidamente si la orden introducida por el usuario es una orden interna del DOS.

Si una orden del usuario no es una orden interna del DOS, el DOS tiene que cargar la orden externa en memoria. Cada vez que el procesador de órdenes carga una orden externa en memoria, pone el contenido del archivo con una extensión EXE o COM en la parte transitoria del procesador de órdenes y en la memoria disponible, si se necesita. Cuando concluye la orden, el espacio que consumía en memoria se libera.

Todas las órdenes del DOS, por tanto siguen unos ciertos pasos de procesamiento. Cada vez que se invoca una orden, el DOS examina primero su lista de órdenes internas. Si la orden se encuentra en memoria, el DOS la ejecuta. En caso contrario el DOS supone que es una orden externa. Para las órdenes externas, el DOS busca primero en el directorio de trabajo en curso un archivo *EXE*, *COM* o *BAT* cuyo nombre coincida con el de la orden introducida. Si el DOS encuentra el archivo correspondiente, lo carga desde el disco en la parte transitoria del procesador de órdenes y en la memoria disponible, si es necesario y, entonces lo ejecuta. En caso contrario, el DOS examina todas las entradas de la ruta que se haya definido mediante la orden *PATH*. Si el DOS no consigue localizar el archivo correspondiente, entonces presenta el mensaje:

Mandato o nombre de archivo erróneo

Cuando la orden externa ha terminado, el DOS libera la memoria reservada para la orden. Estos pasos se muestran en la figura siguiente. Si se está invocando a un procedimiento por lotes (un archivo *BAT*), el procesador de órdenes se limita a leer cada línea del archivo y a llevar a cabo el mismo procedimiento que se ha indicado anteriormente.



Pasos del procesamiento de órdenes del DOS

ANEXO No. 4

ARCHIVOS OCULTOS

Al emitir la orden **CHKDSK** del DOS con su disco del sistema del DOS, se verá lo siguiente:

```
A> CHKDSK B:
```

```
362496 bytes es el espacio total en disco
 37888 bytes en 2 archivo(s) oculto(s)
324608 bytes disponibles en disco
```

```
524288 bytes total en memoria
270992 bytes libres en memoria
```

Los dos archivos ocultos a que se refiere lo anterior son *IBMDOS.DOS* e *IBMBIOS.COM* (PC-DOS) o *MSDOS.SYS* e *IO.SYS* (MS-DOS). Si estos archivos no están presentes en su disco de arranque, el DOS no puede crear su primera y segunda capa y el arranque no tendrá éxito, dando lugar al mensaje:

```
Error en disquete o disquete sin DOS
cámbielo y pulse cualquier tecla
```

Además, si el DOS no puede encontrar el procesador de órdenes (COMMAND.COM, por defecto), entonces se verá el mensaje:

Intérprete de mandatos es erróneo o no se encontró

Colóquese un disco que contenga el procesador de órdenes en la unidad especificada y pulse una tecla. El DOS leerá el contenido del mensaje pasándolo a memoria y completará su tercera capa.

ANEXO No. 5

REGISTRO DE ARRANQUE

El primer sector del disco (pista 0, sector 1, cara 0) contiene el registro de arranque del DOS. El registro de arranque se encuentra en todos los discos del DOS. Esto asegura que los discos del sistema del DOS sean autoarrancables y que los discos que no sean autoarrancables puedan producir el mensaje de error:

Error en disquete o disquete sin DOS
Cámbielo y pulse cualquier tecla para continuar

Cada vez que arranca, el DOS busca en primer lugar el registro de arranque en el disco que está en la unidad A. Si el disco de la Unidad A es autoarrancable, el DOS lo arrancará con él. Si no hay disco en la unidad A, el DOS examinará la partición activa del disco rígido en busca de un registro de arranque.

Además de iniciar el proceso de arranque del DOS, el registro de arranque es responsable de definir la estructura del disco que va a utilizar el DOS como disco del sistema. Considere el registro de arranque que se muestra en la figura siguiente. Los programas que residen en la ROM del computador leen el programa de autoarranque del disco y lo pasan a la memoria cada vez que arranca el computador. La misión principal del programa de autoarranque es cargar los archivos *IBMBIO.COM* e *IBMDOS.COM* (PC-DOS) o *IO.SYS* y *MSDOS.COM* (MS-DOS) en memoria y pasar entonces el control al DOS en la memoria para que continúe el arranque del sistema. Cuando esta parte del DOS recibe el control del computador, comienza la generación del sistema (*SYSGEN*).

Instrucción JMP del 8086
Nombre de IBM o Microsoft y número de versión
Bytes por sector de disco
Sectores por agrupamiento (cluster)
Número de sectores reservados
Entradas máximas en el directorio raíz
Sectores totales
Descripción del medio
Número de sectores por tabla de asignación de archivos
Sectores por pista
Número de cabeza del disco
Número de sectores ocultos
Programa de autoarranque

El registro de arranque del DOS

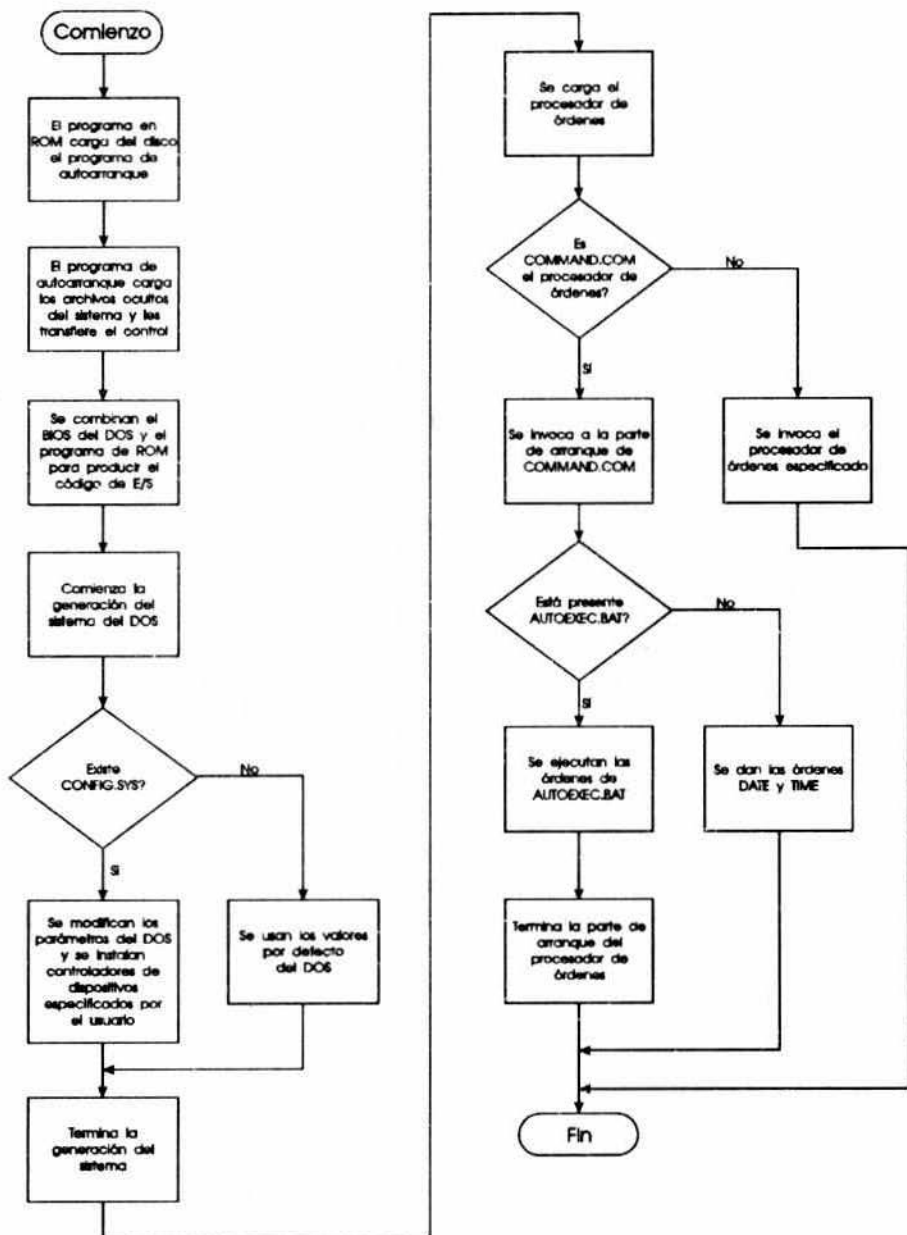
GENERACION DEL SISTEMA

Cuando arranca el DOS, tiene en cuenta los controladores de dispositivos y los parámetros del sistema que se especifican en el archivo *CONFIG.SYS*.

```
FILES=20  
DEVICE=VDISK
```

Si el DOS encuentra a *CONFIG.SYS* en el directorio raíz del dispositivo de arranque, lee el contenido del archivo y utiliza esa información para generar en memoria el sistema operativo. Para el archivo precedente, por ejemplo, el DOS reservará espacio en memoria para 20 llaves de archivo (el valor por defecto es de 8 llaves de archivo) e instalará el controlador de dispositivo para una unidad de disco en RAM. Una vez que el DOS ha examinado el contenido del archivo *CONFIG.SYS*, cierra el archivo y carga el procesador de órdenes desde el disco. Como se ha indicado anteriormente, el DOS utiliza el archivo *COMMAND.COM* por defecto.

Una vez está presente en memoria, la parte de autoarranque del procesador de órdenes busca en el directorio raíz el archivo *AUTOEXEC.BAT* y ejecuta las órdenes que contiene, si el archivo está presente. En caso contrario, lanza las órdenes *DATE* y *TIME*. Este proceso se muestra en la siguiente figura:



Pasos de procesamiento involucrados en la operación de arranque del DOS

ANEXO No. 6

EL SISTEMA OPERATIVO

Es un conjunto de programas que controla, coordina, supervisa y maneja los recursos del computador para el usuario.

Los recursos del computador son las impresoras, unidades de disco, la consola (pantalla y teclado), graficadores, etc.

El sistema operativo puede verse como si fuera una secretaria que maneja una oficina: Los recursos son la máquina de escribir, el teléfono, el archivador, el mensajero; el usuario es el jefe de esta oficina; para él son transparentes los recursos de esta oficina. Por ejemplo, si desea enviar carta de respuesta a un cliente, simplemente la dicta a su secretaria: Escribir la carta en la máquina, elaborar el sobre con la dirección del destinatario, entregarla al mensajero para ponerla al correo, son labores que realiza la secretaria por él.

El Sistema Operativo permite al usuario utilizar con facilidad los programas y aplicaciones y la creación y manejo de los archivos en el computador.





ANEXO No. 7

SISTEMA OPERATIVO EN DISCO DOS

Entre los muchos sistemas operativos para microcomputadores existentes en el mercado, se destaca como uno de los más populares el **DOS (Disk Operating System)**, de la compañía norteamericana productora de software *Microsoft*.

El sistema operativo DOS está diseñado para trabajar con microcomputadores basados en la familia de microprocesadores 8088/8086 de la Intel.

Aunque de este sistema operativo existen las versiones **PC-DOS** e **IBM-DOS** para la IBM, y el **MS-DOS** para otros fabricantes de computadores, su funcionamiento desde el punto de vista del usuario es prácticamente igual de una versión a otra.

Internamente el sistema operativo para microcomputador DOS se divide en dos grandes bloques: el *DOS* y el *BIO* según se muestra en la figura:

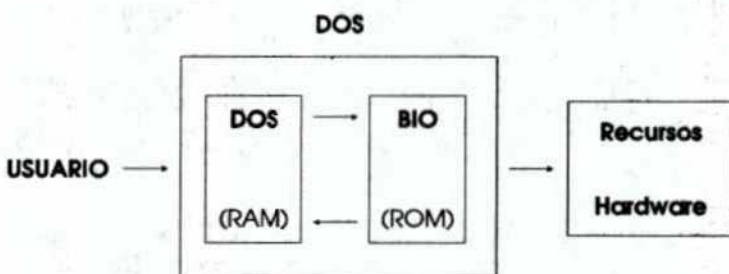


Diagrama de bloques del DOS

El módulo *DOS*, que se carga del disco a la memoria **RAM**, cumple labores de supervisión y lógica; el módulo *BIO* que tiene una parte residente en memoria **ROM**, se encarga de la parte operativa comunicándose y manejando los dispositivos tales como unidades de disco, impresoras y monitores a través de los *drivers* (manejadores lógicos de dispositivos).

Los usuarios del sistema operativo pueden ser programas o personas.

Los programas se comunican con el sistema operativo a través de *interrupciones* y *funciones*; por ejemplo, un lenguaje de alto nivel utiliza llamadas al sistema operativo cuando requiere escribir hacia o, leer desde un archivo en la unidad de disco.

Las personas y algunos programas, se comunican con el sistema operativo mediante *comandos*.

SISTEMA OPERATIVO EN DISCO

NUCLEO DEL DOS

El núcleo del DOS es la parte del DOS responsable de las siguientes tareas:

- Gestión de archivos (creación, borrado o modificación de archivos del DOS).
- Gestión de directorios (creación, borrado o modificación de directorios y de entradas de directorio).
- Interfaz de la aplicación con los servicios del DOS.

Muchos de los usuarios del DOS se refieren a esta sección del DOS como a la "*sección de programa*". Durante el arranque del sistema, el archivo **IBMDOS.COM** (PC-DOS) o **MSDOS.SYS** (MS-DOS) se lee. Las funciones que se ofrecen en esta sección del DOS hacen

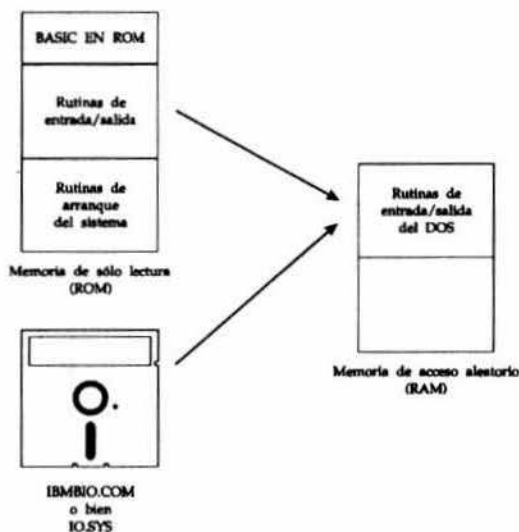
fácil para los programadores desarrollar rutinas potentes sin más que acceder a los servicios incorporados del DOS.

El hecho más importante que hay que recordar acerca del núcleo del DOS es que proporciona una conexión entre nuestros programas y el DOS. Cuando un programa escribe en la pantalla, en el disco o en la impresora, utiliza los servicios ofrecidos por el núcleo del DOS.

DOS (Sistema básico de entrada/salida)

Todos los computadores vienen con un conjunto de rutinas desarrolladas por el fabricante, que le permiten llevar a cabo la entrada y la salida del más bajo nivel. Estas rutinas residen en un área del computador llamada *memoria de sólo-lectura* o **ROM** (Read Only Memory). A diferencia del contenido de la *memoria de acceso aleatorio* **RAM** (Random Access Memory), que se pide cada vez que el computador se desconecta de la red, el contenido de la ROM existe indefinidamente. Cada vez que se conecta el computador, el DOS utiliza las rutinas que se encuentran en la ROM junto con el archivo **IBMBIO.COM** (PC-DOS) o **IO.SYS** (MS-DOS) para construir un área de memoria que se encargue de la entrada y de la salida.

El DOS se comunica con estas rutinas intercambiando paquetes de información. En muchos casos, se necesita un programa llamado controlador de dispositivo (device driver) para traducir el paquete de información de un formato que comprende el DOS a un formato comprensible para el dispositivo físico.



El DOS construye un área de memoria para entrada y salida

Los controladores de dispositivos hacen fácil añadir nuevo material al computador. Como cada dispositivo físico utiliza un formato de paquete conocido y los paquetes del DOS están bien definidos, el programa que controla el dispositivo, necesita asegurarse nada más que el DOS pueda comunicarse con el dispositivo y que todos los mensajes entre el dispositivo material y el DOS se intercambien con éxito. Por ésto es por lo que es necesario especificar al DOS el nombre de cualquier controlador de dispositivo que se necesite (un ratón o un trazador, por ejemplo) mediante **DEVICE = entrada** en el archivo *CONFIG.SYS*.

```
DEVICE=VDISK  
DEVICE=MOUSE
```

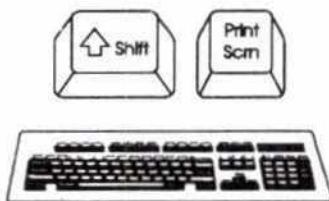
INTERRUPCIONES DEL DOS

Una interrupción es una señal de un programa o dispositivo físico para el procesador. Las interrupciones suspenden momentáneamente la tarea que esté desempeñando la CPU para que pueda desempeñar otra misión. Cada interrupción tiene una sección de código de programa residente en memoria, la cual ejecuta el procesador cada vez que tiene lugar la interrupción. Este código se llama un *gestor de interrupciones* o una *rutina de servicios de interrupciones*.

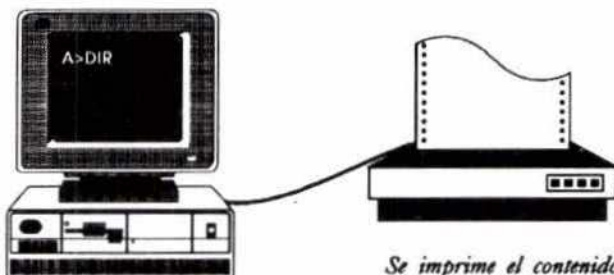
Cuando tiene lugar una interrupción, el procesador localiza al gestor de interrupciones asociado y ejecuta esa sección de código. Por ejemplo, cuando un usuario mantiene pulsadas las teclas **MAYUSC (Shift)** e **IMPR PANT (Print Scrn)** en el teclado, una interrupción suspende momentáneamente el programa que esté corrigiendo en ese momento e imprime el contenido de la pantalla. Cuando el procesador completa la rutina de servicio, devuelve el procesamiento a la tarea que estuviese desarrollando antes de la interrupción, como se ilustra en la página siguiente:



El computador está procesando la orden DIR del DOS



El usuario pulsa las teclas Mayúsc. (Shift) e Impr. Pant. (Print Scrn), produciendo una interrupción



Se imprime el contenido de la pantalla



La orden DIR reanuda el control y continúa

Una interrupción por el retorno a la tarea que se procesaba

El procesador halla los gestores de interrupciones utilizando vectores de interrupción. Se pueden ver los vectores de interrupción como una tabla que contiene el número de la interrupción y la dirección en memoria del gestor de esa interrupción, como se muestra en la tabla siguiente:

Número de interrupción	Dirección
1	0070 01ED
2	3D28 09D2
3	0070 01E2
4	3D28 0998
5	F000 FF54
.	.
.	.
.	.

Vectores de Interrupción del DOS

Cuando tiene lugar una interrupción, el procesador utiliza el número de la interrupción como índice de la tabla para hallar la dirección en memoria del código que hay que ejecutar. Como se ha indicado anteriormente, el núcleo del DOS proporciona la interfaz entre los programas del usuario y del DOS. En la bibliografía recomendada se muestra la forma de usar esta interfaz a través de interrupciones del DOS. Como se verá, es posible generar programas potentes en minutos, utilizando los servicios del DOS.

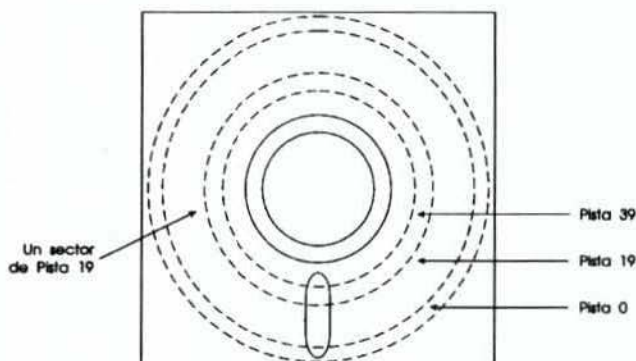
ANEXO No. 8

EL DISCO FLEXIBLE Y LAS UNIDADES DE DISCO FLEXIBLE

El medio de almacenamiento secundario típico en un microcomputador es un disco flexible o disquete.

La información se escribe sobre los disquetes en círculos concéntricos denominados **pistas** (Tracks), cuando la cabeza de lectura/escritura se mueve hacia adentro o afuera de una pista a otra.

Por conveniencia en el manejo de la información, una pista está dividida en secciones denominadas **sectores**. El espacio sobre un sector se mide en **bytes**. En un byte se puede almacenar un carácter.



El disquete

El número de pistas, sectores y bytes, depende del tipo de disquete. La siguiente sección describe los tipos de unidades de disco y de disquetes.



ANEXO No. 9

TIPOS DE UNIDADES DE DISCO

La tabla muestra los tipos de unidades de disco que se pueden manejar con la versión 3.30 de DOS.

Tamaño (pulgadas)	Descripción	Capacidad (bytes)
5 1/4	Un lado	160 Kb / 180 Kb
5 1/4	Dos lados	320 Kb / 360 Kb
5 1/4	Alta capacidad	1.2 Mb
3 1/2	Dos lados	720 Kb
3 1/2	Dos lados	1.44 Mb

Tipos de unidades de disco

ANEXO No. 10

TIPOS DE DISQUETES

La siguiente tabla muestra los tipos de disquete utilizados para leer y escribir (grabar) información.

Tamaño (pulg)	Descripción	Pistas	Sectores	Capacidad (bytes)
5 1/4	Un lado, doble densidad	40	8/9	160 Kb / 180 Kb 512 Bytes/Sector
5 1/4	Dos lados, doble densidad	40	8/9	320 Kb / 360 Kb 512 Bytes/Sector
5 1/4	Alta capacidad, doble densidad	80	15	1.2 Mb 512 Bytes/Sector
3 1/2	Dos lados	80	9	720 Kb 512 Bytes/Sector
3 1/2	Dos lados	80	18	1.44 512 Bytes/Sector

Tipos de disquete

En general, una unidad de disco puede leer y escribir sobre disquetes formateados en otra unidad de igual o menor capacidad del mismo tamaño; por ejemplo, una unidad de disco de 5 1/4" y 1.2 MB alta capacidad, puede leer/escribir sobre discos formateados en una unidad de 5 1/4" y 360 KB doble cara, doble densidad; pero lo contrario no es cierto.

ANEXO No. 11

PREPARACION DE DISCOS NUEVOS

Antes de poder utilizar un disquete nuevo, es necesario prepararlo escribiéndole el formato (pistas, sectores, Bytes/sector), utilizado por DOS. Para ello se utiliza el comando **FORMAT** que se describe en la sección "*Comandos del DOS*".

ANEXO No. 12

PROTECCION DE DISCOS

Debido a que es posible escribir accidentalmente sobre un disco, borrando su información, los disquetes poseen un medio para prevenir la escritura, permitiendo solamente leerlos.

- Los discos de 5 1/4" tienen una muesca, la cual, cuando está *cubierta* pone el disquete en **estado de protección contra escritura** (write-protect).
- Por otro lado, los discos de 3 1/2" tienen en la esquina inferior derecha, una ventana de **write-protect**, la cual, cuando está *abierta*, impide que se pueda escribir sobre el disquete.



ANEXO No. 13

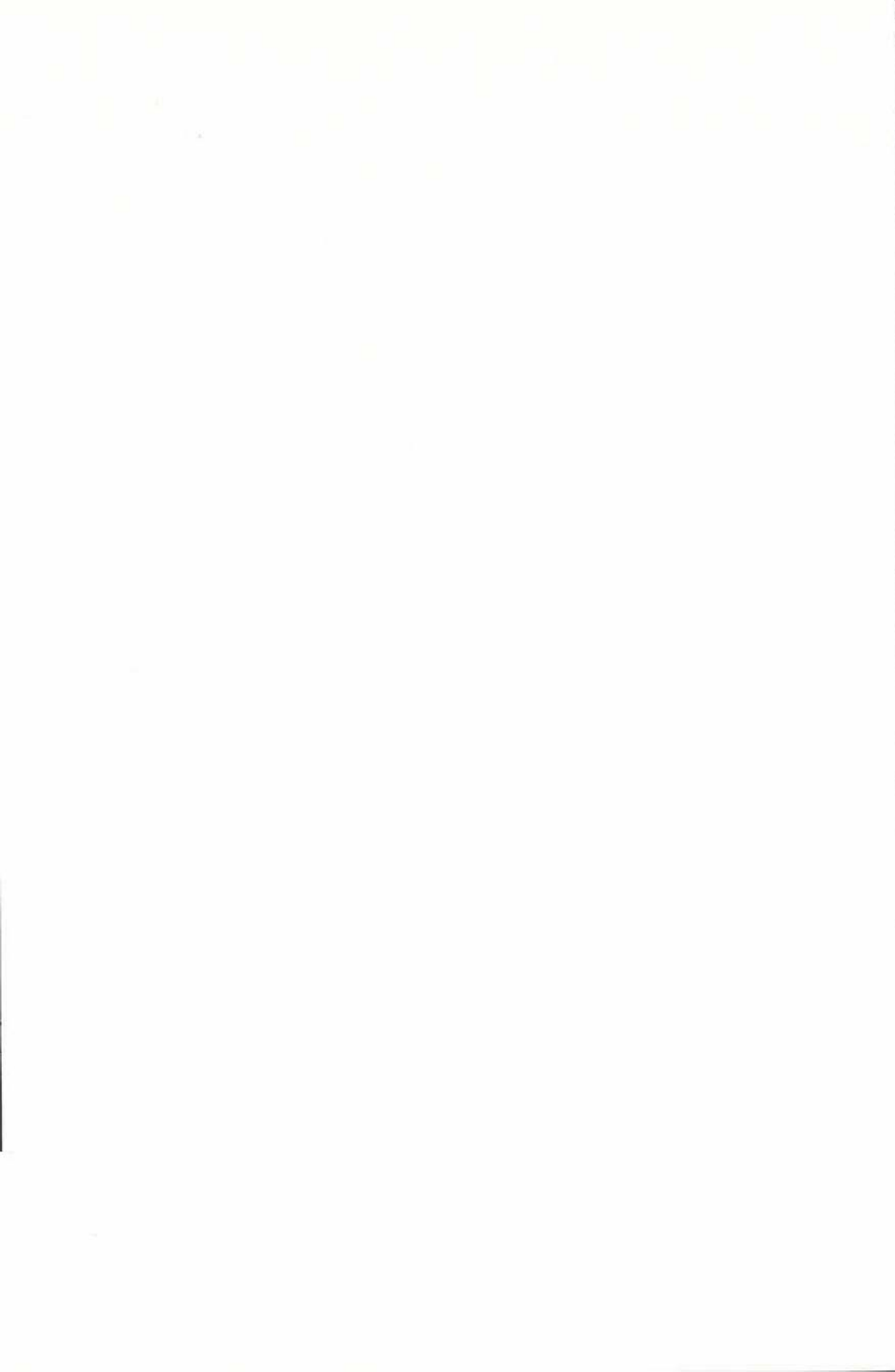
NOMBRES DE DISPOSITIVOS DOS

Algunos dispositivos que pueden utilizarse con el computador son las pantallas, impresoras, unidades de disco flexible y unidades de disco duro. Estos dispositivos son reconocidos por DOS por nombres específicos. Estos *nombres de dispositivos-DOS*, se deben conocer y emplear cuando se desee utilizarlos.

La tabla siguiente lista algunos de tales nombres. Para obtener completa información de los nombres de los dispositivos, consulte su manual de referencia DOS.

Dispositivo	Nombre DOS de dispositivo
Impresora	LPT o PRN
Display o Consola	CON (Consola)
Unidades de Disco - drives (disquete o disco duro)	Las letras A - Z seguidas de dos puntos (:) Ej.: A:

Dispositivos DOS



ANEXO No. 14

UNIDADES DE DISCO

Un computador puede venir equipado con unidades de disco flexible, unidades de disco duro o una combinación de ambas.

DOS se refiere a ellas por medio de letras empezando con la **A**, en orden ascendente hasta la **Z**.

Cuando se prende un equipo, DOS verifica qué clase de unidades están conectadas y les asigna una letra como identificador a los discos flexibles y, a continuación, a los discos duros.

Las letras **A** y **B** están reservadas para unidades de disco flexibles y la letra **C** para el primer disco duro; por ejemplo, si un computador tiene una unidad de disco flexible y una de disco duro, asignará la letra **A** al disco flexible y la **C** al disco duro.

En la práctica, la letra identifica al *disco*, no a la unidad de disco.

» Note que **unidad de disco** se refiere al "tocadoisco".



ANEXO NO. 15

DISCOS FLEXIBLES

Uno de los factores más importantes que afectan la calidad de la organización de sus discos, son los rótulos o etiquetas que se colocan en ellos. **No se debe escribir nunca en la etiqueta de un disco, una vez que está colocada en él.** Si el disco tiene ya una etiqueta y hay que cambiarla, despegue la etiqueta o coloque una nueva encima de ella.

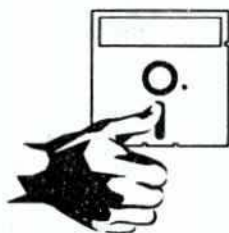
REGLAS PARA MANEJAR LOS DISCOS FLEXIBLES



*No deje nunca el disco
cerca de un dispositivo
magnético*



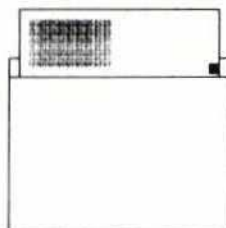
*Mantenga el disco alejado
del teléfono*



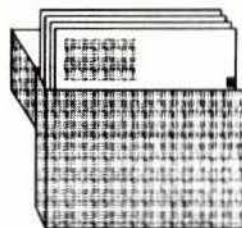
*No toque nunca la
superficie de sus discos
flexibles*



*No fume nunca junto a
sus discos flexibles*



Guarde los discos en sus sobres, cuando no los utilice.



Guarde sus discos flexibles en lugar seguro.



Haga siempre copias de seguridad de sus discos flexibles.



Manténgalos a temperatura entre 10 y 44 grados centígrados.



Nunca doble los discos flexibles.

La abertura de lectura/escritura permite a la cabeza de lectura/escritura que hay dentro de la unidad de disco, acceder al medio de almacenamiento. Cuando el disco gira en la unidad de disco, la cabeza de lectura/escritura que hay dentro de la unidad, accede a la información contenida en el disco a través de esta abertura. El agujero índice se usa para la temporización a medida que gira el disco dentro de la unidad. Tenga cuidado de no tocar el medio de almacenamiento a través de ninguna de estas aberturas.

ANEXO No. 16

FUNCIONES DOS

El computador está dotado de un conjunto de teclas que, por sí solas, constituyen una función específica. A continuación se describen estas teclas:

Función	Teclado del Computador Personal	Teclado del Computador AT - PS/2
Cancela Comando	Ctrl - Break	Ctrl - Break
Pausa en Pantalla	Ctrl - Num Lock	Pause
Imprime Pantalla	Shift - PrtSc	PrtSc
Eco. en Impresora	Ctrl - PrtSc	Ctrl - PrtSc
Reinicia Sistema	Ctrl - Alt - Del	Ctrl - Alt - Del

Funciones DOS

Las columnas dos y tres, se refieren a nombres de teclas, no a palabras; así por ejemplo, **Ctrl - PrtSc** significa presionar la tecla **Control** y sostenerla mientras se pulsa la tecla **PrintScreen**.

CANCELA COMANDO

Cancela la ejecución de un comando u orden que viene trabajando normalmente. También se llama algunas veces "*terminar un programa*".

PAUSA EN PANTALLA

Hace que el despliegue de la información en la pantalla se detenga cuando está apareciendo con mucha rapidez para leerla. Para continuar, se pulsa cualquier tecla.

**IMPRIME
PANTALLA**

Imprime (en la impresora), la información que está presente actualmente en la pantalla. La impresora debe estar prendida y en línea, previamente.

**ECO EN
IMPRESORA**

Imprime la línea actual de la pantalla, cada vez que se pulsa la tecla **ENTER**. Para apagar esta acción, presione el **Eco** de nuevo.

**REINICIA
SISTEMA**

Reinicia el DOS. (Hace que se cargue de nuevo el DOS desde el disco a la memoria).

ANEXO NO. 17

ARCHIVOS DOS

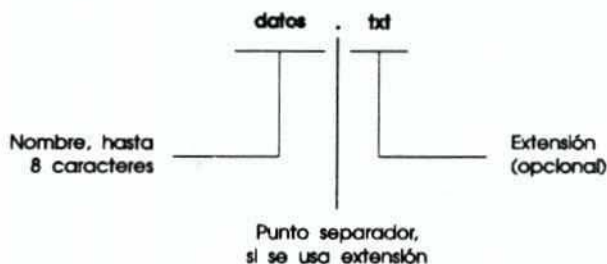
Un archivo consiste en un conjunto de información relacionada, la cual normalmente se guarda en un disquete o en un disco duro.

Un archivo se identifica con un nombre. Cuando se desea que DOS busque en un archivo, se le da su nombre, lo que permite leer o escribir información en él.

NOMINACION DE ARCHIVOS

La nominación de archivos se rige por las siguientes reglas:

1. El nombre completo de un archivo consta de *nombre* + *extensión*. Ejemplo:



2. Cada archivo debe tener un nombre único: No puede haber dos archivos con nombre igual. Sin embargo, puede haber archivos con idéntico nombre en distintos discos o en diferentes directorios en el disco duro.
3. El nombre debe tener de 1 a 8 caracteres de longitud.
4. La extensión es opcional; si se usa, debe tener de 1 a 3 caracteres.
5. No se debe utilizar en la nominación ninguno de los siguientes caracteres:

< > . : + ? * | / \ " [] ; =

ESPECIFICACION DE ARCHIVOS

Para localizar un archivo en un disco es necesario, además del nombre y extensión, indicar la unidad de disco donde se encuentra. La letra que identifica la unidad de disco, debe ir acompañada de *dos puntos* (:). Ejemplo:

A:datos.txt

La letra y los dos puntos se conocen como el *especificador* del archivo.

- » No puede haber espacios entre el especificador, el nombre y la extensión.

ANEXO No. 18

INICIACION DEL DOS

Iniciar el DOS significa leer los comandos internos desde un disquete que contenga DOS y colocarlos en la memoria del computador.

El sistema operativo DOS estará listo para recibir comandos, cuando pregunta por la fecha y hora actual.

INICIACION DIARIA DEL DOS

Dependiendo del caso específico, se selecciona uno de los siguientes procedimientos:

INICIACION DEL DOS DESDE UN DISQUETE

Si el computador está apagado:

1. Inserte el disco que contiene los archivos del DOS en la unidad de disco A.
2. Conecte la impresora (si tiene una), el monitor de video y luego el computador.

3. Espere un momento mientras el sistema se verifica y carga por sí mismo. La espera dependerá de la cantidad de memoria que posea el equipo.

Si el computador está prendido:

1. Inserte el disco que contiene los archivos del DOS en la unidad de disco A.
2. Haga una reiniciación del Sistema (presione **Ctrl-Alt-Del**). La luz de actividad de la unidad de disco parpadeará indicando que el DOS se está leyendo desde el disco.
3. Espere un momento mientras el sistema se carga por sí mismo.

INICIACION DEL DOS DESDE UN DISCO DURO

El Sistema Operativo DOS se iniciará desde un disco duro en cualquiera de los siguientes casos:

- Al conectar el computador.
- Al hacer una reiniciación del Sistema presionando **Ctrl-Alt-Del**.

NOTA: La puerta de la unidad de disco flexible A, deberá estar abierta cuando se haga cualquiera de los dos pasos anteriores.

INICIACION DE LA FECHA Y HORA DEL SISTEMA

INICIACION DE LA FECHA

Cuando se ha cargado el DOS y está listo, se verá un mensaje similar a éste:

```

La fecha actual es Mie  4-04-1990
Introduzca nueva fecha: (mm-dd-aa): _
    
```

El cursor indica que DOS está esperando una respuesta. Si la fecha es correcta, presione **ENTER**. Si no, escriba la fecha correcta y termine con **ENTER**.

En el mensaje anterior:

- mm** Significa mes. Escriba dos números entre *01* y *12*.
- dd** Indica día. Digite uno o dos números entre *01* y *31* para el día correspondiente.
- aa** Indica año. Escriba dos números entre *80* y *99* o los cuatro números entre *1980* y *1999*.

Separe las partes con *guión (-)*, *slash (/)*, o *punto (.)*

Ejemplos:

1-31-90 ó 1/31/90 ó 1.31.90

1-31/1990 ...

INICIACION DE LA HORA

Después de entrar la fecha, DOS solicita la hora:

```
La hora actual es 21:11:30.19
Introduzca nueva hora: _
```

El cursor indica que DOS está esperando una respuesta. Si la hora es correcta, presione **ENTER**. Si no, escriba la hora correcta y termine con **ENTER**.

En el ejemplo anterior el formato es:

```
hh:mm:ss.cc
```

Donde:

hh Indica hora.

mm Indica minutos.

ss Indica segundos.

cc Indica centésimas de segundos.

Ejemplos:

16:34:04.29

17:35:01

19.30

NOTA: El uso de *guiones* (-) o *slash* (/) no opera. Para una explicación más amplia sobre la hora, consulte el comando **TIME** en el anexo siguiente.

ESTADO DE PREPARADO DEL DOS

Después que se ha entrado la fecha y la hora, DOS muestra un mensaje similar al siguiente, dependiendo del computador que se esté usando:

```
The IBM Personal Computer DOS
Version 3.30 (C)Copyright IBM Corp 1981, 1987
          (C)Copyright Microsoft Corp 1981, 1986
```

A>

A> es el *prompt* o *puntero* de DOS. Si se inicia DOS desde un disco duro, el prompt será C>. El prompt significa que DOS está listo para recibir información, es decir, está esperando que se le digiten comandos.

ESPECIFICACION DE LA UNIDAD DE DISCO POR DEFECTO

El prompt A> indica la unidad de disco sintonizada. Cualquier letra que aparezca en el prompt, es la unidad por defecto. El sistema DOS buscará en esta unidad los archivos o comandos que se digiten, a no ser que se le indique otra unidad en la especificación de archivo.

Cuando se requiera, puede cambiarse la unidad de disco por defecto, tecleando la nueva designación de unidad seguida de dos puntos (:). Por ejemplo:

A>	Prompt original.
A> B:↵	Nueva designación de unidad.
B>	Nuevo prompt.

En adelante, **B** es la unidad en la cual DOS buscará cuando se digite un comando o nombre de archivo, a no ser que se indique otra unidad en la especificación de archivo.

ANEXO No. 19

COMANDOS DEL DOS

TIPOS DE COMANDOS DOS

Cuando se carga el DOS al inicio de una sesión de trabajo, o cuando se reinicia con **Ctrl-Alt-Del**, DOS carga los *comandos internos* y el archivo *COMMAND.COM* y los coloca en la memoria del computador.

Estos comandos son llamados *internos* porque permanecen en la memoria del computador hasta que se apaga el equipo o se reinicia el DOS nuevamente. Por este motivo, no se necesita tener colocado el disco con los archivos del sistema operativo permanentemente, para utilizar los comandos internos. Ejemplo de *comandos internos*, son los comandos **DIR** y **COPY**.

El resto de los comandos sobre el disquete DOS se denominan *comandos externos* porque no son leídos en la memoria del computador cuando se carga DOS; por consiguiente, cada vez que se requiera utilizar estos comandos, se deberá colocar el disco con el sistema operativo en la unidad **A** o **B** para leerlos desde allí.

Cualquier archivo que tenga la extensión **COM**, **EXE**, **BAT**, se considera un *comando externo* (por ejemplo *FORMAT.COM* y *COMP.COM* son comandos externos).

Cuando se selecciona un comando externo no hay necesidad de incluir la extensión en el nombre del archivo.

CONSIDERACIONES UTILES PARA DIGITAR COMANDOS

- Espere hasta que aparezca el *prompt* de DOS, **A>**

NOTA: En estos Anexos se supone que la unidad de disco por defecto es **A**, a no ser que se especifique lo contrario.

- Digite el comando y las otras partes que requiera (por ejemplo, el especificador de la unidad de disco o una especificación de archivo). Se pueden utilizar letras mayúsculas, minúsculas (o una combinación de ellas), por ejemplo:

Dir A:

Utilice espacios para separar las partes del comando.

- No separe las tres partes de una especificación de archivo; por ejemplo:

A:datos.txt

» es una especificación correcta.

A: datos txt

» es incorrecta.

- Para cancelar un comando que se está ejecutando, presione **Ctrl-Break**. (Esto sólo funciona cuando el comando está leyendo el teclado o está escribiendo en la pantalla).

NOTACION UTILIZADA EN LA DESCRIPCION DE LOS COMANDOS

En la notación utilizada para la descripción de los comandos DOS de la sección siguiente, se debe tener en cuenta:

- [] Los ítemes encerrados entre paréntesis cuadrados son opcionales. Si se incluyen, no se escribirán los paréntesis. Por ejemplo, el comando DIR tiene esta sintaxis:

```
DIR [d:][ruta][nombre archivo[.ext]][/P][W]
```

Al utilizarlo sin opciones se digita:

DIR

Si se desea con la opción P, se digita:

DIR /P

Mayúsculas

Las palabras mostradas en mayúsculas se conocen como *palabras claves*. Los nombres de los comandos DOS son palabras claves. Al digitarlas pueden escribirse en mayúsculas o minúsculas.

Minúsculas

Los ítemes en minúsculas significan que el usuario deberá sustituirlos. Por ejemplo, en la sintaxis del comando **DIR** mostrada arriba, **nombre de archivo** indica que el usuario deberá escribir el nombre de su archivo en vez de la palabra **nombre de archivo**.

- ... Los puntos suspensivos indican que se puede repetir un ítem. Por ejemplo, en el comando **PRINT** cuya sintaxis es

```
PRINT [[d:]nombre archivo[.ext]][/T][C][P] ...
```

se puede tener este uso:

Print a:Nomina, a:Carta, b:notas.txt

Incluya en el comando toda la puntuación que se muestre tal como *comas* (,), *signo igual* (=), *interrogaciones* (?), *asteriscos* (*), *dos puntos* (:), *slashes* (/), y *backslashes* (\). Por ejemplo, el comando **TIME** cuya sintaxis es:

TIME [hh[:mm[:ss[:cc]]]]

se puede usar así:

TIME 16:34:04.29

ANEXO No. 20

ARCHIVOS DEL DOS

Cuando se almacenan datos en un disco, el DOS sitúa los datos en una zona de almacenamiento que se llama *archivo*. Los archivos de disco se tratan de la misma forma que se utilizan los archivos de su armario archivador. Los archivos de disco se crean, se modifica su contenido, se les cambia de nombre y, por último, se eliminan cuando ya no se necesitan.

Todos los archivos del DOS tienen un nombre. Los nombres de los archivos tienen dos partes: *Un nombre de archivo y una extensión*. El *nombre del archivo* debe contener entre 1 y 8 caracteres. Los caracteres admisibles son las letras del alfabeto junto con los caracteres

~ ! @ # \$ % ^ & () - _ { } ' ,

La *extensión* es opcional y puede contener hasta tres caracteres. Un punto separa el nombre del archivo de la extensión, como sigue:

nombrarch.ext

Lo que sigue son nombres válidos de archivos del DOS:

COMMAND.COM
NOMINAS.OCT
PRESUPUE
CIENCIAS.!!!

VENTAS.SEP
INFORME.LIB
MI NOTA
SISTEM\$1.DAT



Estos nombres son de archivo incorrecto:

DEMASIADOS_CARACTERES

<i>.EXE</i>	(no tiene nombre de archivo).
<i>LIBRO..EXT</i>	(el doble punto es un caracter inaceptable).
<i>BLANCO .EXE</i>	(el espacio es inaceptable).
<i>DATOS.NUEVOS</i>	(la extensión sólo puede tener tres letras).

El DOS tiene varios nombres reservados para dispositivos tales como la impresora. Como el DOS ya está usando los nombres siguientes, no se pueden crear archivos con los mismos nombres:

<i>AUX</i>	<i>CON</i>	<i>LPT3</i>
<i>CLOCK\$</i>	<i>LPT1</i>	<i>NUL</i>
<i>COM1</i>	<i>LPT2</i>	<i>PRN</i>
<i>COM2</i>		

ANEXO No. 21

COMANDO DIR

La orden de directorio del DOS, **DIR**, nos permite enumerar los nombres y atributos de los archivos contenidos en un disco.

Al dar la orden **DIR**, DOS responderá con algo como ésto:

Volumen en unidad A sin etiqueta
Directorio en A:\

COMMAND	COM	23210	3-11-86	10:00a
ASSIGN	COM	1509	8-11-86	10:00a
ATTRIB	EXE	15091	3-11-86	10:00a
FORMAT	COM	6831	5-09-86	3:52p
FDISK	COM	14448	5-13-86	9:47a
DISKCOMP	COM	3774	5-20-86	8:52a
DISKCOPY	COM	4099	5-20-86	8:52a
TREE	COM	6306	5-03-86	11:19a
EXE2BIN	EXE	2816	3-11-86	10:00a
FC	XC	14576	3-11-86	10:00a
FIND	EXE	6403	3-11-86	10:00a
JOIN	EXE	15971	3-11-86	10:00a
LABEL	EXE	2750	3-11-86	10:00a
SHARE	EXE	8304	3-11-86	10:00a
SUBST	EXE	16611	3-11-86	10:00a
SYS	EXE	2671	3-11-86	10:00a
MORE	COM	282	3-11-86	10:00a
SORT	EXE	1664	3-11-86	10:00a
CHKDSK	COM	9435	3-11-86	10:00a
DEBUG	COM	15552	3-11-86	10:00a
EDLIN	COM	7261	3-11-86	10:00a
PRINT	COM	8291	3-11-86	10:00a
RECOVER	COM	4050	3-11-86	10:00a
MODEM	COM	2868	5-21-86	11:59a
ANSI	SYS	2563	5-12-86	1:42p

SELECT	COM	6052	5-08-86	8:54a
KEYBSP	COM	2317	5-08-86	6:10p
KEYBFR	COM	2395	5-08-86	6:13p
KEYBGR	COM	2310	5-08-86	6:15p
KEYBNR	COM	2335	5-08-86	6:10p
KEYBSF	COM	2331	5-08-86	6:10p
KEYBSG	COM	2360	5-08-86	6:20p
KEYBD	COM	2335	5-08-86	6:25p
KEYBU	COM	1972	5-08-86	6:08p
KEYBIT	COM	2011	5-08-86	6:06p
GFTPRN	COM	292	3-18-86	10:30a
GRAFTABL	COM	2145	3-11-86	10:00a
GRAPHICS	COM	5363	4-10-86	11:37a
COMP	COM	7194	5-03-86	11:19a
SHIPZONE	EXE	779	5-12-86	9:45a
SPEED	EXE	1196	5-09-86	11:34a
		42 archivo(s)	54272 bytes libres	

La salida de la orden **DIR** informa varias cosas. En primer lugar, la línea:

```
Volumen en unidad A sin etiqueta
Directorio en A:\
```

dice qué unidad contiene el disco que **DIR** está enumerando. En segundo lugar, **DIR** indica el nombre, el tamaño (en bytes) y la fecha y hora de creación para todos los archivos, como se ilustra aquí:

```
Volumen en unidad A sin etiqueta
Directorio de A:\

COMMAND      COM      23210      11-3-86      10:00a
```


Por último, **DIR** dice cuántos bytes libres quedan en el disco. El formato de la orden de directorio del DOS es:

```
DIR [espec_arch] [/P] [/W]
```

en donde ocurre lo siguiente:

espec_arch

es la especificación de archivo (*el nombre*) del archivo que hay que citar. Si no se proporciona una especificación de archivo, **DIR** presentará todos los archivos del disco.

/P

indica a **DIR** que se detenga cada vez que haya una pantalla llena de información y que exponga el mensaje:

```
Pulse cualquier tecla para continuar...
```

» Pulse simplemente una tecla y el listado del directorio continuará.

/W

indica a **DIR** que muestre solamente los nombres de los archivos que están presentes en el disco, suprimiendo la fecha y hora de creación de todos los archivos, con el objeto de presentar varios nombres de archivos en cada línea.

Considere los ejemplos siguientes:

DIR /W

DIR suprime el tamaño, así como la fecha y hora de creación de todos los archivos. Sólo muestra los nombres de los archivos. Digite:

DIR COMMAND.COM

DIR sólo muestra información acerca del archivo *COMMAND.COM*. Si **DIR** no puede encontrar el archivo que se especifica, presentará el mensaje:

No se encontraron archivos.

Estando seleccionado el drive **A** como unidad por defecto, la orden del DOS:

DIR FORMAT.COM

Produce como resultado:

Volumen en unidad **A** sin etiqueta
Directorio en **A:**

FORMAT	COM	11458	28-05-86	12:00a
	1 Archivo(s)	54272 bytes	libres	

ANEXO NO. 22

CARACTERES "COMODIN" DEL DOS

Una de las mejores características del DOS es que permite sustituir los caracteres * y ? por otros caracteres en el nombre de un archivo durante las operaciones de búsquedas de archivos. El signo de *interrogación* (?) en un nombre de archivo le dice al DOS que esa posición del nombre del archivo puede estar ocupada por cualquier caracter. Si se digita el ejemplo siguiente:

DIR CAPIT_?TXT

estamos pidiendo al DOS que proporcione un listado de directorio con todos los archivos cuyos nombres empiezan por la palabra *CAPIT_* seguida por cualquier caracter y que tiene la extensión *TXT*. Si se aplica la orden anterior al disco que contiene los anexos de este libro, el DOS mostrará:

Volumen en unidad A sin etiqueta
Directorio de A:\

CAPIT_1	TXT	23210	3-11-86	10:00a
CAPIT_2	TXT	1509	3-11-86	10:00a
CAPIT_3	TXT	6831	5-09-86	3:52p
CAPIT_4	TXT	14448	5-13-86	9:47a
CAPIT_5	TXT	3774	5-20-86	8:52a
CAPIT_6	TXT	4099	5-20-86	8:52a

Al dar la siguiente orden de directorio en su disco del sistema del DOS:

DIR DISK????.COM

El DOS mostrará lo siguiente:

Volumen en unidad A sin etiqueta

Directorio de A:\

DISKCOPY	COM	3774	5-20-86	8:52a
DISKCOM	COM	4099	5-20-86	8:52a
2 Archivo(s)		54272 bytes libres		

Los cuatro signos de interrogación le dicen al DOS que nos importan los cuatro últimos caracteres del nombre del archivo, siempre y cuando los cuatro primeros sean *DISK* y la extensión sea *COM*. El caracter comodín * es más potente que el signo de interrogación. No solamente le dice al DOS que la posición en cuestión puede estar ocupada por cualquier caracter, sino que además le dice al DOS qué caracteres quedan en el nombre del archivo o en la extensión. Considere los ejemplos siguientes. Transcriba la orden siguiente y el DOS le mostrará todos los archivos cuya extensión sea *COM*, independientemente del nombre del archivo:

DIR A:*.COM

Transcriba la orden siguiente y **DIR** le mostrará todos los archivos cuyos nombres empiecen por *DISK* y que tengan la extensión *COM*:

DISK*.COM

Transcriba la orden siguiente y **DIR** le nombrará todos los archivos cuyos nombres empiecen por la letra *D* y que tengan la extensión *COM*:

DIR D*.COM

Transcriba la orden siguiente y **DIR** le mostrará todos los archivos del disco:

DIR*.*

- » Recuerde que el asterisco equivale no sólo a los caracteres que se encuentren en la posición que está, sino también a los posibles caracteres que estén a la derecha.

Los caracteres-comodín del DOS son una posibilidad que se va a utilizar diariamente. Tómese su tiempo ahora para familiarizarse con su utilización.

CONVENCIONES DEL DOS PARA NOMBRAR ARCHIVOS

Todos los archivos del DOS almacenan una de dos cosas: Programas o datos. Normalmente se usa la extensión de tres caracteres del nombre de archivo para diferenciar los tipos de archivos. Como los archivos del DOS pueden contener algunas cosas, utilice las extensiones para indicar lo que contiene el archivo. Lo que sigue son algunos ejemplos que se usan frecuentemente:

<i>COM</i>	Ordenes del DOS o programas ejecutables.
<i>EXE</i>	Programas ejecutables del DOS.
<i>BAT</i>	Archivos por lotes del DOS.
<i>SYS</i>	Controles de dispositivos que se pueden instalar.
<i>TXT</i>	Archivos de procesamiento de textos.
<i>DAT</i>	Archivos de datos para programas.

Examinando las extensiones del nombre de archivo, se pueden clasificar los archivos como se muestra en la tabla:

Nombre	Misión
<i>COMMAND.COM</i>	Orden del DOS
<i>ANSI.SYS</i>	Controlador de dispositivo ANSI
<i>CAPIT.TXT</i>	Documento de procesador de textos
<i>DATOS.SEP</i>	Archivo de datos
<i>PRESUP.EXE</i>	Programa ejecutable

Clases de archivo del DOS

Es importante que los nombres de los archivos tengan significado para usted. Por ejemplo, los archivos que contienen los anexos de esta cartilla, se denominaron como sigue:

ANEXO_1.TXT
ANEXO_2.TXT
ANEXO_3.TXT
.
.
.
ANEXO_43.TXT

ANEXO No. 23

COMANDO CHKDSK

La orden **CHKDSK** del DOS proporciona varios datos útiles acerca del disco concreto y de la utilización actual de la memoria.

El formato de la orden **CHKDSK** es:

`[unidad:] [camino]CHKDSK[espec_arch] [/F] [/V]`

donde ocurre lo siguiente:

Unidad Es la unidad de disco que contiene la orden **CHKDSK.COM**. Si no se especifica una unidad, el DOS utilizará la unidad por defecto en curso.

Camino Es el camino del directorio de la orden **CHKDSK.COM**. Si no se proporciona un camino, el DOS empleará el directorio por defecto en curso.

Espec_arch Es el nombre del archivo cuya contigüidad en disco tiene que examinar el DOS. Los caracteres comodín son útiles como especificación de archivo para examinar la contigüidad del disco.

La contigüidad hace referencia a la forma como un archivo está contenido en un disco: Si un archivo ocupa varios *clusters* es posible que dichos clusters no estén uno a continuación del otro sino esparcidos en diferentes partes del disco, Si éste es el caso, DOS gastará más tiempo para manipular el archivo

- /F** Indica a **CHKDSK** que corrija los errores que encuentre en la tabla de asignación de archivos o en los directorios. Las correcciones se escriben en el disco. De vez en cuando el DOS pierde uno o más de los punteros internos que emplea para llevar la cuenta de los archivos. La opción **/F** indica a **CHKDSK** que repare tantos punteros dañados, como le sea posible.
- /V** Indica a **CHKDSK** que visualice los nombres y caminos de todos los archivos del disco.

CHKDSK proporciona información acerca de lo siguiente:

- Etiqueta de volumen del disco (si la hay).
- Espacio total en disco (bytes).
- Espacio consumido por archivos ocultos (bytes).
- Espacio consumido por directorios (bytes).
- Espacio consumido por archivos del usuario (bytes).
- Espacio consumido por sectores dañados (bytes).
- Espacio disponible en el disco (bytes).
- Memoria total presente (bytes).
- Memoria total disponible (bytes).

CHKDSK supone que el disco que se desea examinar ya está contenido en la unidad de disco especificada. A diferencia de otras órdenes del DOS, **CHKDSK** no le pedirá que coloque el disco en la unidad específica. Suponiendo que no se especificase una unidad de disco a examinar, como en:

CHKDSK

CHKDSK emplearía la unidad por defecto en curso.

La primera línea que presenta **CHKDSK** muestra el rótulo del volumen del disco. Los rótulos de volumen se estudian detenidamente más adelante en este anexo. Si el disco no tiene rótulo de volumen, **CHKDSK** se limita a omitir toda la línea. Las

seis líneas siguientes ofrecen información sobre la utilización del disco. Las dos últimas líneas tratan del uso de memoria por el sistema. Como se ilustra, hay 524.388 bytes de memoria presentes en el sistema. El DOS reserva varios de miles de bytes al arrancar el sistema, lo cual deja disponible 485.328 bytes.

Para enumerar todos los archivos contenidos en un disco, digite la orden:

CHKDSK/V

CHKDSK responderá con:

Directorio A:\
 A:\IO.SYS
 A:\MSDOS.SYS
 A:\COMMAND.COM
 A:\ASSIGN.COM
 A:\ATTRIB.EXE
 A:\FORMAT.COM
 A:\FDISK.COM
 A:\DISKCOPY.COM
 A:\TREE.COM
 A:\EXE2BIN.EXE
 A:\FC.EXE
 A:\FIND.EXE
 A:\JOIN.EXE
 A:\LABEL.EXE
 A:\SHARE.EXE
 A:\SUBST.EXE
 A:\SYS.EXE
 A:\MORE.COM
 A:\SORT.EXE
 A:\CHKDSK.COM
 A:\DEBUG.COM
 A:\EDLIN.COM
 A:\PRINT.COM
 A:\RECOVER.COM
 A:\MODE.COM
 A:\ANSI.SYS
 A:\SELECT.COM
 A:\KEYBSP.COM

A:\KEYBFR.COM
A:\KEIBGR.COM
A:\KEIBNR.COM
A:\KEIBSF.COM
A:\KEIBSG.COM
A:\KEIBDK.COM
A:\KEYBUK.COM
A:\KEYBIT.COM
A:\KETBSV.COM
A:\KFTPRN.COM
A:\GFTPRN.COM
A:\GRAFTABL.COM
A:\GRAPHICS.COM
A:\COMP.COM
A:\SHIPZONE.EXE
A:\SPEED.EXE

362496 bytes en el espacio total en disco

39936 bytes en 2 archivo(s) oculto(s)

268288 bytes en 42 archivo(s) de usuario

54272 bytes disponibles en disco

524288 bytes total en memoria

274080 bytes libres de memoria

Si su disco se usa mucho, indique a **CHKDSK** que corrija los punteros dañados como sigue:

CHKDSK/F

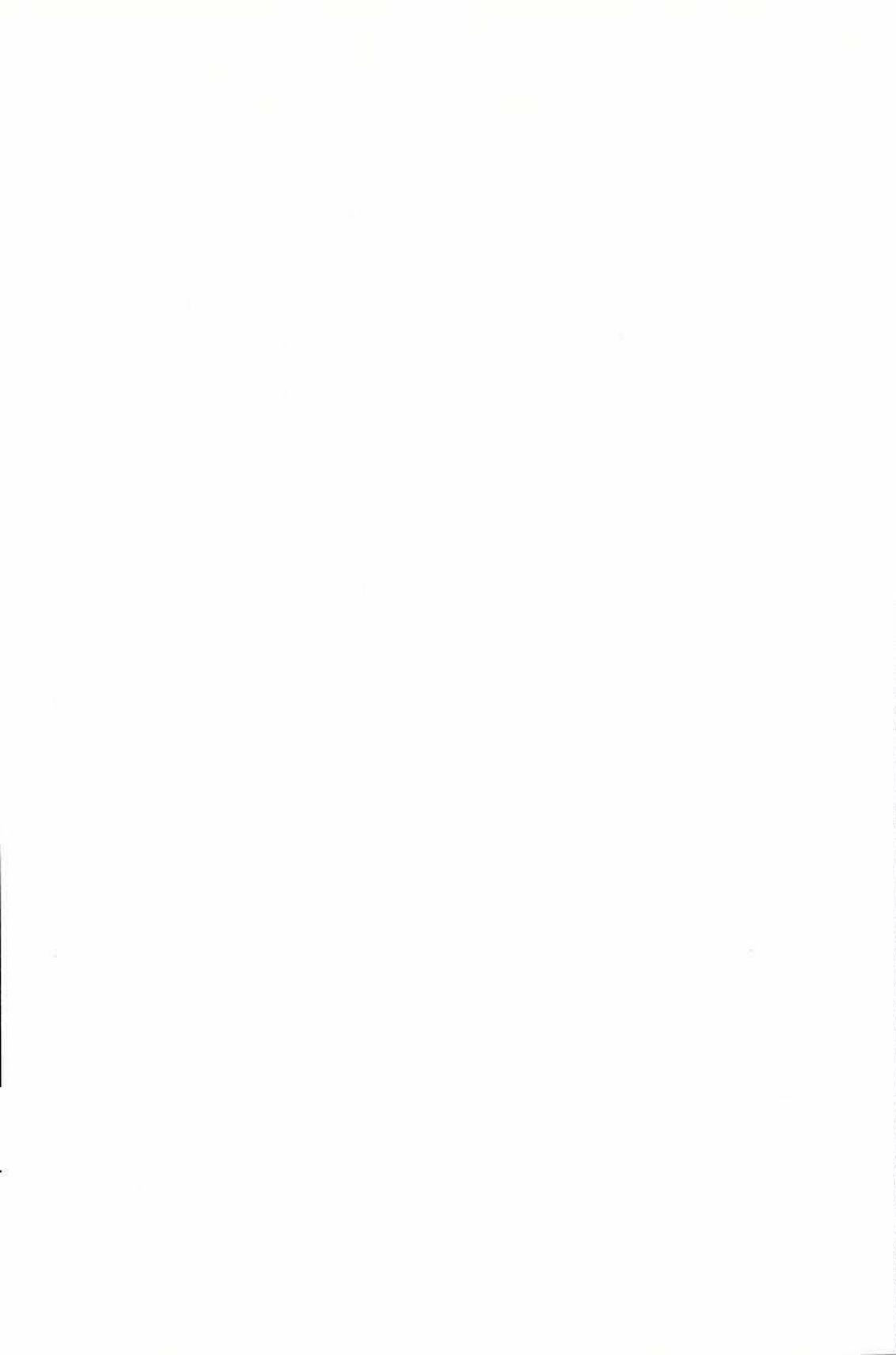
Si **CHKDSK** encuentra algún puntero dañado, responderá con:

xxx clusters perdidos encontrados en xxx cadenas.
Convertir cadenas perdidas en archivos (S/N)?

Si se pulsa **S**, **CHKDSK** empezará a crear archivos en el directorio raíz con el nombre **FILEnnnn.CHK**, donde **nnnn** son números consecutivos que empiezan por el **0000**. El primer archivo que creará **CHKDSK** será **FILE0000.CHK**, el segundo será

FILE0001.CHK, y así sucesivamente. Examine todos los archivos que crea **CHKDSK** y borre los que no sean utilizables.

Un factor de gran importancia en lo tocante a la velocidad de *E/S* en disco es la contigüidad en el disco. Tómese su tiempo para leer cuidadosamente este anexo. Es importante revisar periódicamente la contigüidad del disco. Al hacerlo, notará que sus aplicaciones corren con mucha más eficiencia.



ANEXO No. 24

COMANDO COPY

El comando interno **COPY** se usa para duplicar archivos y enviar datos hacia o desde dispositivos de entrada/salida.

El formato de la orden es:

```
COPY [d1:][ruta1][nombre archivo1[.ext]][/A][/B][d2:]  
[ruta2][nombre archivo2[.ext]][/A][/B][/V]
```

Donde:

[d1:][ruta1][nombre archivo1[.ext]] es el archivo de **origen** (dispositivo de entrada).

[d2:][ruta2][nombre archivo2[.ext]] es el archivo de **destino** (dispositivo de salida).

- /A** hace que el archivo sea tratado como un archivo *ASCII* (textos). El final de la marca de archivo (*filemark*) determina dónde termina el archivo.
- /B** hace que el archivo sea tratado como un archivo binario. El final del archivo se determina por el tamaño del archivo indicado en el directorio.
- /V** verifica que los datos escritos en el archivo de destino estén correctos.



ANEXO No. 25

COMANDO COMP

El comando externo **COMP** compara dos archivos. **COMP** se usa principalmente después del comando **COPY** para asegurar que ambos archivos son idénticos.

El formato de la orden es:

```
COMP [d1:][ruta1][nombre archiv1[.ext]][d2:][ruta2]  
[nombre archiv2][.ext]]
```

Donde:

[d1:][ruta1][nombre archiv1[.ext]] es el nombre del archivo primario. Se pueden comparar archivos múltiples usando los caracteres globales * y ?.

[d2:][ruta2][nombre archiv2[.ext]] es el nombre secundario. Si no se escribe el nombre del archivo secundario, sólo se comparan los archivos con un nombre igual al del archivo primario.

Lo primero que **COMP** compara es el tamaño de los archivos. Si el tamaño es diferente, la comparación termina. **COMP** continúa mostrando otras diferencias hasta que encuentre 10.

EJEMPLO DEL COMANDO

COMP TESTDATA.TXT SAFEDATA.BAK

compara el contenido del archivo *TEST.DATA* con *SAFEDATA.BAK*.

```
Comparación de archivos ok
¿Desea comparar más archivos(S/N)?S
```

Como se puede ver, el contenido de los archivos era idéntico. **COMP** le avisa para entrar otro archivo cuando termina la comparación.

```
Teclee la especificación del archivo primario
GOODFILE.TXT
Teclee la especificación del archivo secundario
BADFILE.TXT
```

No es necesario especificar el nombre del archivo secundario para la comparación, si es igual al primario, sólo la identificación de la unidad de disco o la ruta del directorio. DOS buscará un archivo que tenga un nombre igual al del archivo primario.

```
Comparación de archivos ok
¿Desea comparar más archivos(S/N)?N
A>
```

COMP C:\PROGRAMS\TUTOR.BAS A:TUTOR.BAK

Error de comparación en la posición B10

Archivo 1 = 46

Archivo 2 = 4D

Desea Comparar más archivos (S/N)?N

Aquí se compara el archivo *TUTOR.BAS* en el directorio *PROGRAMS* de la unidad de discos con *TUTOR.BAK* en el directorio corriente de la unidad de discos y se han encontrado diferencias.



ANEXO No. 26

COMANDO DISKCOPY

DISKCOPY es un comando externo que se usa para copiar todo el contenido de un disquete (origen) en otro (destino). Su formato es:

DISKCOPY [d1:] [d2:] [/1]

Donde:

[d1:] [d2:] son las unidades de discos de origen y destino respectivamente. Si no se especifican **d1:** y **d2:** toda la copia será desde/hasta la unidad corriente pre-asignada. Esto se conoce como copia de unidad sencilla. Si se asigna la misma unidad a **d1:** y a **d2:** también se conoce como copia de unidad sencilla.

/1 hace que **DISKCOPY** copie sólo el primer lado del disquete de origen en el primer lado del disquete de destino.

Automáticamente se hace el formato del disquete de destino, si no lo tiene. **DISKCOPY** destruye cualquier información existente en el disquete de destino.

ANEXO No. 27

COMANDO DISKCOMP

La orden **DISKCOMP** del DOS compara el contenido de dos discos del DOS y visualiza los números de cara y de pista que difieren. El formato de la orden es:

```
[unidad:] [camino]DISKCOMP [d1:[d2:]] [/1]/8]
```

Donde ocurre lo siguiente:

Unidad es la unidad de disco que contiene la orden **DISKCOMP.COM**. Si no se especifica una unidad de disco, el DOS utilizará la unidad por defecto actual.

Camino es la ruta del directorio de la orden **DISKCOMP.COM**. Si no se proporciona un camino el DOS empleará el directorio por defecto en curso.

d1: d2: son las unidades de disco que contienen los discos que debe comparar **DISKCOMP**. Si no especifican dos unidades de disco, **DISKCOMP** le pedirá que introduzca el disco. Si tiene un sistema con una sola unidad, **DISKCOMP** le pedirá que vaya poniendo los discos en la unidad de disco en los momentos oportunos.

/1 indica a **DISKCOMP** que compare sólo la primera cara de los discos (la mayoría de los discos es de doble cara).

/8 indica a **DISKCOMP** que emplee ocho sectores por pista (la mayoría de los discos utiliza 9 sectores por pista).

DISKCOMP nos permite usar la misma unidad de disco o unidades distintas para la comparación. Si se usa la misma unidad de disco, el programa le pedirá que introduzca los discos en los momentos oportunos. Suponga que la unidad por defecto es A:

DISKCOMP

Inserte el PRIMER disquete en la unidad A:

Pulse cualquier tecla para continuar...

Comparando 40 pistas
9 sectores/pista, 2 cara(s)

Inserte el SEGUNDO disquete en la unidad A:

Pulse cualquier tecla para continuar...

DISKCOMP llevará a cabo una comparación con una sola unidad, bajo las siguientes condiciones:

- Que el usuario omita ambos parámetros de unidad.
- Que el sistema tenga solamente una unidad de disco flexible.

Si se omite la segunda unidad de disco, **DISKCOMP** usa la unidad por defecto en curso. Si los discos que se están comparando son idénticos, **DISKCOMP** responderá:

Comparación OK

Comparar otro disquete (S/N)?

Si **DISKCOMP** encuentra diferencias entre los discos, presentará el mensaje siguiente:

Error de comparación en cara n, pista nn

DISKCOMP sólo funciona con discos flexibles. Si se utiliza para comparar un disco rígido o virtual, el sistema responderá:

**Especificación de unidad errónea
La unidad especificada no existe, o no es removible.**

Una de las aplicaciones más útiles de **DISKCOMP** es verificar la ejecución de una orden **DISKCOPY** digitando:

**DISKCOPY A: B:
DISKCOMP A: B:**

Si el **DISKCOPY** ha tenido éxito, **DISKCOMP** no debería encontrar ninguna diferencia.



ANEXO No. 28

COMANDO FORMAT

FORMAT es un comando externo que prepara un disco nuevo con un formato aceptable por DOS y le crea un directorio de manera que pueda recibir archivos DOS. Su formato es:

```
FORMAT [d:][/S][/1][/8][/V]
```

Donde:

- d** es el nombre de la unidad de discos que contiene el disco al cual se le va a preparar el formato.
- /S** hace que **FORMAT** copie los archivos del sistema *IO.SYS*, *MSDOS.SYS* y *COMMAND.COM* en el disco al que se le preparó el formato.
- /1** prepara el formato a discos con un sólo lado. Los disquetes con un sólo lado tienen 160KB ó 180KB, dependiendo del número de sectores.
- /8** prepara el formato con 8 sectores por pista en vez del número preasignado que es 9.
- /V** hace que **FORMAT** haga una pausa y pregunte por una etiqueta opcional para el volumen.

Las opciones **/1** y **/8**, existen para guardar compatibilidad con unidades de disco de modelos anteriores.

ANEXO No. 29

COMANDO SYS

El comando externo **SYS** copia en el disquete de la unidad de discos especificada los archivos del sistema almacenados en el disco por defecto.

Los archivos del sistema deben ser los primeros en ser transferidos a un disco vacío o debe haber espacio reservado para ellos en el disquete al cual se deseen transferir. Su formato es:

SYS d:

Donde:

d: es el nombre de la unidad de discos que contiene el disquete a donde se desean transferir los archivos DOS.

NOTA: Tenga cuidado con **SYS**; se podría escribir sobre la información existente si no hay suficiente espacio reservado en el disquete para los archivos del sistema.

Los archivos del sistema que se transfieren son *IO.SYS* y *MSDOS.SYS*.

IO.SYS y *MSDOS.SYS* son archivos ocultos. Esto es, no serán mostrados cuando se pide el listado del directorio con el comando **DIR**.

ANEXO No. 30

COMANDO TYPE

La orden **TYPE** del DOS nos permite ver el contenido de un archivo en la pantalla. El formato de la orden es:

```
TYPE espec_arch
```

Donde ocurre lo siguiente:

espec_arch

es el nombre del archivo que hay que ver en la pantalla. **TYPE** no admite caracteres comodín en la especificación del nombre de archivo.

El DOS solamente permite que **TYPE** muestre archivos de texto (ASCII) en la pantalla. Los archivos que tienen las extensiones de nombre de archivo *COM*, *EXE* u *OBJ* contienen caracteres no imprimibles (no alfanuméricos) que impiden su visualización. Supongamos que se intenta utilizar **TYPE** para ver el contenido de un archivo que no es de texto digitando:

TYPE COMMAND.COM

La pantalla se llenará de caracteres extraños, y es probable que el computador empiece a emitir pitidos. Esto se debe a los caracteres no imprimibles.

Supongamos que el archivo *ESTADOS.DAT* contiene lo siguiente:

WASHINGTON
ARIZONA
COLORADO
NEVADA
UTAH
NUEVA YORK

La orden:

TYPE ESTADOS.DAT

mostrará lo siguiente:

WASHINGTON
ARIZONA
COLORADO
NEVADA
UTAH
NUEVA YORK

Como ya se ha indicado con anterioridad, **TYPE** no admite los caracteres comodín. Al utilizar caracteres comodín, **TYPE** responderá con:

Nombre de archivo no válido o no se encontró.

Si se intenta ver un archivo que no existe **TYPE** responderá:

No se encontraron archivos.

TYPE sí permite ver archivos que estén contenidos en un disco distinto del disco por defecto en curso. Por ejemplo, supongamos que el archivo *CIUDADES.DAT* que hay en la unidad **B** contiene:

SEATTLE
PHOENIX
COLORADO SPRINGS
LAS VEGAS
SALT LAKE CITY
NUEVA YORK CITY

Digite la orden

TYPE B:CIUDADES.DAT

La orden mostrará:

```

SEATTLE
PHOENIX
COLORADO SPRINGS
LAS VEGAS
SALT LAKE CITY
NUEVA YORK CITY
    
```

Además, **TYPE** permite ver archivos contenidos en otros subdirectorios.



ANEXO No. 31

USO DE LOS NOMBRES DE DISPOSITIVOS DEL DOS

Se ha dicho que el DOS reserva nombres para todos estos dispositivos. La lista siguiente proporciona los nombres que reserva el DOS para nombres de dispositivos:

AUX	El primer puerto asíncrono de comunicaciones (se usa para comunicaciones en serie).
COM1	También es el primer puerto asíncrono de comunicaciones (se usa para comunicaciones en serie).
COM2	El segundo puerto asíncrono de comunicaciones (se usa para comunicaciones en serie).
CLOCK\$	El reloj de tiempo real del sistema.
CON	Para entrar, CON apunta al teclado; para salida, apunta a la pantalla.
LPT1	El primer puerto paralelo de impresora.
LPT2	El segundo puerto paralelo de impresora.
LPT3	El tercer puerto paralelo de impresora.
PRN	El primer puerto paralelo de impresora.

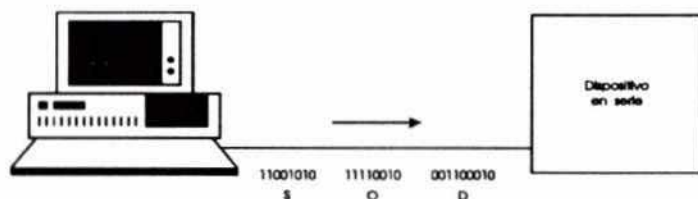
COMUNICACION DE DATOS EN SERIE Y EN PARALELO

Varios de los dispositivos que acabamos de enumerar se han clasificado en serie o en paralelo. Es importante entender las diferencias entre la comunicación de datos en serie y en paralelo. Cuando el computador se comunica con la mayoría de los dispositivos periféricos, lo hace de dos maneras: Con comunicación en serie o con comunicación en paralelo. El computador envía datos a los dispositivos periféricos caracter a caracter. Cada caracter que envía al computador está representado por ocho dígitos binarios o bits. El computador enviará la palabra **DOS** a la impresora como la cadena de dígitos binarios que se muestra en la tabla:

Caracter	Representación binaria
D	01000100
O	01001111
S	01010011

Los dígitos binarios del DOS

Los dispositivos en serie utilizan un cable a lo largo del cual se transmiten los datos. Cada caracter de una palabra está representado por ocho *bits* (dígitos binarios), que se envían por el único cable caracter a caracter, como se muestra en la siguiente figura:

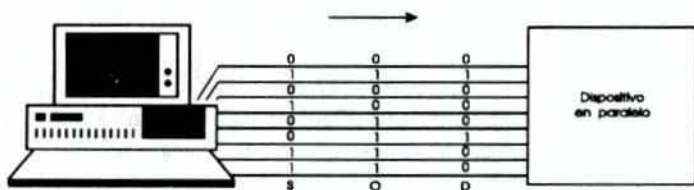


Dispositivo en serie

La comunicación de datos en serie requiere además bits adicionales empotrados, que aseguran la coordinación de los dos dispositivos.

Por el momento, sin embargo, sólo es importante entender que la comunicación en serie tiene lugar bit a bit a lo largo de un único cable.

Los dispositivos en paralelo utilizan ocho cables para transmitir datos, lo cual significa que se pueden enviar los ocho bits en una palabra de una sola vez, como lo indica la siguiente figura:



Comunicación de datos en paralelo

Hay cables adicionales que proporcionan la coordinación entre los dispositivos serie y paralelo que se comunican. Como la comunicación de datos en paralelo tiene lugar de ocho en ocho bits, es mucho más rápida que la comunicación de datos en serie. La mayoría de las impresoras utiliza hoy día comunicación de datos en paralelo. Los modems utilizan comunicación de datos en serie.

REFERENCIAS A NOMBRES DE DISPOSITIVOS DEL DOS

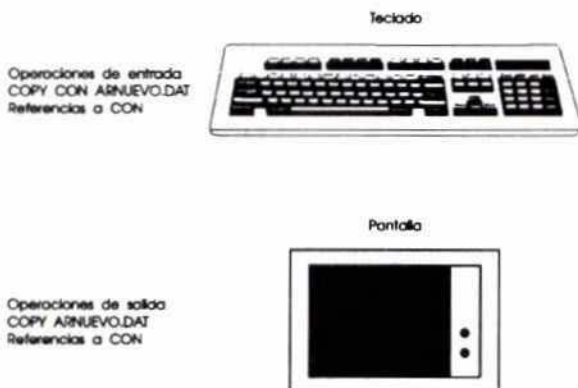
La posibilidad de hacer referencia a los nombres de dispositivos del DOS resulta útil en el funcionamiento diario del DOS. Dé la siguiente orden:

COPY CON ARNUEVO.DAT

Su pantalla debería mantener ahora:

COPY CON ARNUEVO.DAT

Como se ha afirmado anteriormente, el nombre de dispositivo **CON** apunta al teclado durante la entrada y a la pantalla durante la salida, según muestra la figura siguiente:



Referencias a CON

En este caso, el DOS colocará cualquier cosa que digite en el teclado, dentro del archivo *ARNUEVO.DAT*.

Transcriba lo siguiente:

**Para la entrada CON apunta al teclado.
Para la salida CON apunta la pantalla.**

Una vez digitadas las frases, su pantalla debería contener:

```
A>COPY CON ARNUEVO.DAT
Para la entrada CON apunta al teclado.
Para la salida CON apunta la pantalla.
```

Pulse la tecla de función F6 y entonces pulse ENTER. La pantalla debería contener:

```
A>COPY CON ARNUEVO.DAT
Para la entrada CON apunta al teclado.
Para la salida CON apunta a la pantalla.
^Z
1 Archivo (s) copiados
```

El DOS utiliza el ^Z (Ctrl-Z) para marcar el final de los archivos que sólo contienen texto. La tecla F6 nos permite insertar el ^Z al final del archivo que estamos creando desde el teclado.

Dé la orden:

TYPE ARNUEVO.DAT

El DOS responderá con:

```
Para la entrada CON apunta al teclado.
Para la salida CON apunta a la pantalla.
```

Encienda la impresora y dé la orden:

COPY ARNUEVO.DAT LPT1:

En este caso, el DOS copiará el contenido del archivo *ARNUEVO.DAT* en la impresora. Dé la orden:

COPY CON LPT1:

Digite lo siguiente:

**Para la entrada CON apunta al teclado.
LPT1: Es el primer puerto de impresora.**

Pulse las teclas **F6** y **ENTER**. Su pantalla debería contener ahora:

```
A COPY CON LPT1:
Para la entrada CON apunta al teclado.
LPT1: es el primer puerto de impresora.
^Z

1 Archivo(s) copiado(s)
```

Las frases que acaba de introducir deberían empezar a imprimirse. El dispositivo **LPT1**: apunta al primer puerto paralelo del sistema. Normalmente este puerto está conectado a la impresora del sistema. El nombre de dispositivo **PRN**: también se refiere a la impresora del sistema. Si se da la orden

COPY ARNUEVO.DAT PRN:

Esta acción también imprimirá el contenido de *ARNUEVO.DAT*.

- » Obsérvese que usar **PRN**: suele ser más seguro que usar **LPT**: porque **PRN**: se refiere al primer puerto de impresora conectado (que puede ser **LPT2**:). La orden **PRINT** del DOS que se presenta más adelante, aumentará la utilidad de su impresora.

ANEXO No. 32

COMANDO PRINT

PRINT es un comando externo que controla la secuencia de impresión de los archivos de texto. Una característica sobresaliente de **PRINT** es que permite que continúen entrando por el teclado comandos adicionales, mientras **PRINT** está enviando textos a la impresora.

El formato de la orden es:

```
PRINT [[d:]nombre archivo[.ext]][/T][/C][/P] ...
```

Donde:

[[d:]nombre archivo[.ext]] es el nombre del archivo que se desea imprimir. El archivo debe estar en el directorio corriente, no se permite usar rutas. Se puede poner hasta 10 archivos en la cola de impresión.

- /T** cancela los archivos de la cola y detiene la impresión inmediatamente.
- /C** permite remover archivos individuales de la cola de impresión.
- /P** alinea los archivos en la cola de impresión (prepara una cola) para imprimirlos. Es la opción preasignada, si no se especifica alguna otra.

Cada opción especificada permanece activa hasta que se especifique otra.

PRINT usado solo, muestra el estado de la cola de impresión.

ANEXO No. 33

REDIRECCION DE LA ENTRADA Y DE LA SALIDA

Las órdenes DOS muestran sus resultados en la pantalla. Por ejemplo, consideremos la orden:

DIR

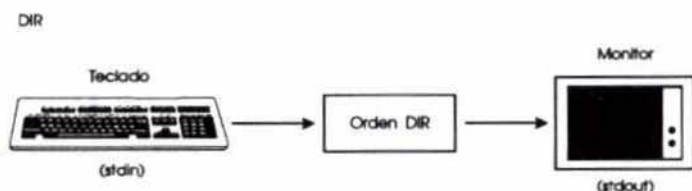
El DOS muestra la salida de la orden de directorio en la pantalla. La mayor parte de las órdenes del DOS envía su salida a un destino definido por el DOS como salida estándar (*stdout*). Por defecto, *stdout* señala a la pantalla.

Este anexo presenta los operadores de redirección de *E/S* (*entrada/salida*) del DOS. Como se verá, la misión de estos operadores es simplemente la de redirigir la *E/S*, cambiándola con respecto a la salida estándar por defecto (la pantalla) y la entrada estándar (el teclado).

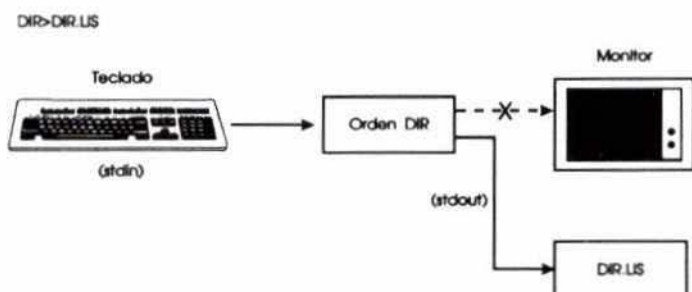
Con la orden siguiente:

DIR > DIR.LIS

la salida de la orden **DIR** no se ve en la pantalla. En lugar de esto, el operador de redirección **>** del DOS indicó al DOS que colocase la salida de la orden en el archivo *DIR.LIS*, como se ilustra en las siguientes figuras:



Vía de comunicación de la orden DIR



Redirección de la comunicación utilizando DIR > DIR.LIS

Se puede dirigir la salida de las órdenes del DOS a archivos, a la impresora (PRN:), e incluso a otros programas. Dé la orden siguiente:

DIR *.BAT > PRN:

Si la impresora del sistema está conectada, se imprimirá la salida del listado del directorio.

La orden siguiente:

CHKDSK ESTADO.DISK

colocará la salida de la orden **CHKDSK** en el archivo **ESTADO.DISK**.

El segundo operador de redirección del DOS, > >, añade la salida de una orden al archivo especificado. Considere la orden siguiente:

DIR *.SYS > > SYS.DIR

Cuando el DOS ejecuta la orden, busca el archivo **SYS.DIR**. Si el archivo existe, el DOS añade el resultado de la orden al final del archivo. Si éste no existe, el DOS lo crea y coloca en él la salida de la orden.

DIR *.SYS > > SYS.DIR

Si en el disco están los archivos **ANSI.SYS** y **VDISK.SYS**, el archivo **SYS.DIR** contiene ahora:

Volumen en unidad A sin etiqueta
Directorio en A:\

ANSI	SYS	1641	14/08/84	8:00
UNDISK	SYS	3080	14/08/84	8:00
2 Archivo(s)		9024 bytes libres		

Para verificar ésto, se digita la orden siguiente:

TYPE SYS.DIR

Los dos operadores de redirección que se han examinado hasta el momento se refieren estrictamente a la redirección de la salida. El DOS también nos permite redirigir la entrada, cambiándola con respecto a la entrada estándar (*stdin*). Las tres órdenes del DOS que se usan aquí son **MORE**, **FIND** y **SORT**.

MORE

La orden **MORE** toma su entrada de la entrada estándar y la muestra, pantalla a pantalla, por la salida estándar. Cada vez que se muestra una pantalla de información, **MORE** se detiene y presenta el mensaje siguiente:

-- Más --

MORE espera hasta que se pulsa una tecla para continuar. Si desea ver más salida, pulse simplemente una tecla. Si no, pulse **CTRL BREAK**.

Digite la orden siguiente para redirigir la salida y crear el archivo *DIR.LIS*:

DIR > DIR.LIS

Ahora puede ver el contenido del archivo por pantallas como sigue:

MORE < DIR.LIS

MORE mostrará la primera página de datos y pedirá al usuario que pulse una tecla para continuar, como se muestra aquí:

Volumen en unidad A sin etiqueta

Directorio de A:\

COMMAND	COM	22042	14/08/84	8:00
ANSI	SYS	1641	14/08/84	8:00
SORT	EXE	1632	14/08/84	8:00
SHARE	EXE	8544	14/08/84	8:00
FIND	EXE	6363	14/08/84	8:00
ATTRIB	EXE	15123	14/08/84	8:00
MORE	COM	320	14/08/84	8:00
ASSIGN	COM	988	14/08/84	8:00
PRINT	COM	7811	14/08/84	8:00
SYS	COM	3629	14/08/84	8:00
CHKDSK	COM	9275	14/08/84	8:00
FORMAT	COM	9015	14/08/84	8:00
VDISK	COM	3080	14/08/84	8:00
BASIC	COM	17024	14/08/84	8:00
BASICA	COM	26880	14/08/84	8:00
FDISK	COM	8076	14/08/84	8:00
COMP	COM	3471	14/08/84	8:00
TREE	COM	2473	14/08/84	8:00
BACKUP	COM	5440	14/08/84	8:00
RESTORE	COM	5413	14/08/84	8:00

-- Más --

Cuando se pulsa una tecla, **MORE** muestra la página siguiente (si existe una página completa) o hasta el final del archivo.



ANEXO No. 34

COMANDO RENAME

RENAME (o **REN**) es un comando interno que permite cambiar el nombre de un(os) archivo(s).

El formato de la orden es:

```
REN [d:][ruta]nombre archivol[.ext] nombre
archivo2[.ext]
```

Donde:

[d:][ruta]nombre archivol[.ext] es el nombre del archivo o archivos a los que se les desea cambiar el nombre. Se puede cambiar el nombre a múltiples archivos usando los caracteres globales * y ?

nombre archivo2[.ext] es el nuevo nombre del archivo.

- » Note que no se puede cambiar el nombre a un archivo y ponerlo en otro directorio, por lo tanto no se debe especificar la ruta.

EJEMPLO DEL COMANDO

REN C:\PROGRAMS\TUTOR.BAS XYZZY.TXT

Este comando cambia el nombre del archivo *TUTOR.BAS* del directorio *PROGRAMS* en la unidad de discos *C* a *XYZZY.TXT*. No se especifica la ruta del directorio con el nuevo nombre del archivo.

REN DATOVIEJ DATONUEV

Este comando simplemente le cambia el nombre al archivo *DATOVIEJ* del directorio corriente, a *DATONUEVO*.

REN *.BAS *.BAK

Este comando cambia el nombre a todos los archivos en el directorio corriente que terminen en *BAS* para que terminen en *BAK*.

ANEXO No. 35

COMANDO DEL

El comando interno **DEL** o **ERASE** elimina archivos del disco duro o del disquete.

El formato de la orden es:

```
DEL [d:][ruta][nombre archivo[.ext]]
```

o también

```
ERASE [d:][ruta][nombre archivo[.ext]]
```

EJEMPLO DEL COMANDO

DEL C:\PROGRAMS\TUTOR.EXE

Elimina el archivo *TUTOR.EXE* del directorio de programas del disco C:.

DEL *.TXT

Elimina todos los archivos con la extensión *TXT* del directorio corriente por defecto.

DEL *.*

Elimina todos los archivos del directorio corriente. **¡Tenga cuidado!**
Este comando puede borrar el disco completo. Para prevenir un borrado accidental, DOS muestra el mensaje:

¿Está seguro (S/N)?

ANEXO No. 36

GESTION DE ARCHIVOS DOS CON SUBDIRECTORIOS

La mayor parte de los usuarios del DOS se encuentran con que al usar el DOS a diario se produce rápidamente un gran número de archivos en sus discos rígidos o flexibles. Para mejorar la calidad de la organización del disco, el DOS ofrece los subdirectorios. Considere el ejemplo siguiente:

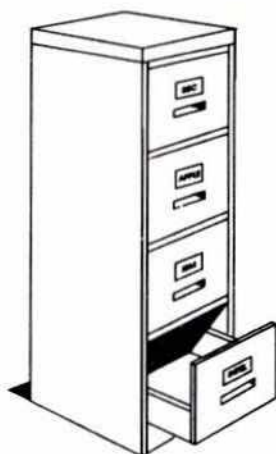
Un agente de cambio y bolsa está al tanto de los precios de las acciones de las cuatro compañías de computadores siguientes:

Apple
DEC (Digital Equipment Corporation)
IBM (International Business Machines)
Intel

Para mejorar la organización de la oficina, el agente adquiere un armario archivador de cuatro cajones, con cada cajón etiquetado para la compañía correspondiente, como se muestra en la figura de la página siguiente.

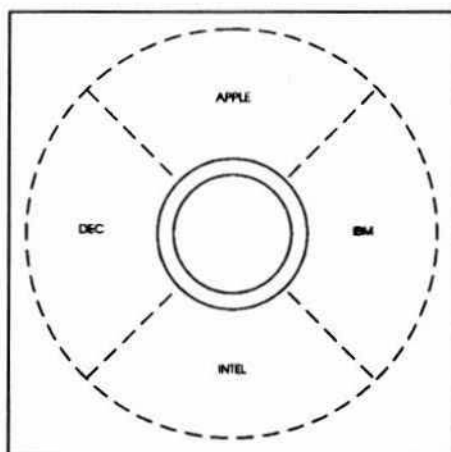
Todos los días el agente añade los informes del cajón apropiado y, como consecuencia, el nivel de organización de su oficina ha mejorado drásticamente.

Los subdirectorios del DOS le permiten llevar a cabo el mismo tipo de organización en el disco. Imagine los subdirectorios del DOS como si fuesen el equivalente de uno de los cajones que hay en el armario archivador del agente de cambio y bolsa.



Archivador para organizar los precios de las acciones

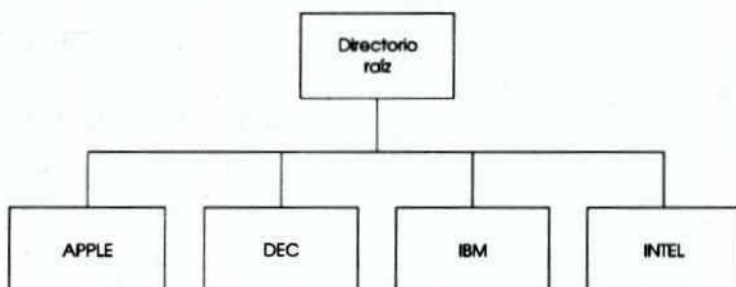
Al poner los archivos de cada compañía en sus respectivos subdirectorios, puede ver solamente los archivos de un grupo específico, cada vez que lleve a cabo un listado del directorio. Esto, a su vez, mejora en la organización de su disco. Además, al tener los archivos en subdirectorios por separado, se disminuye la posibilidad de borrar accidentalmente un archivo. Como se verá, la mayoría de las órdenes del DOS no afectan a los archivos contenidos en otros subdirectorios, a no ser que se haga referencia explícita a un archivo por su nombre.



Subdirectorios correspondientes a los cajones del archivador

COMPRENDIENDO LOS SUBDIRECTORIOS DEL DOS

Por defecto, cada disco del DOS tiene un directorio raíz o de nivel superior. El directorio raíz proporciona la base (o raíz), a partir de la cual parten todos los demás directorios. Si se considera el ejemplo del agente de bolsa se puede ver la estructura de directorios de disco.



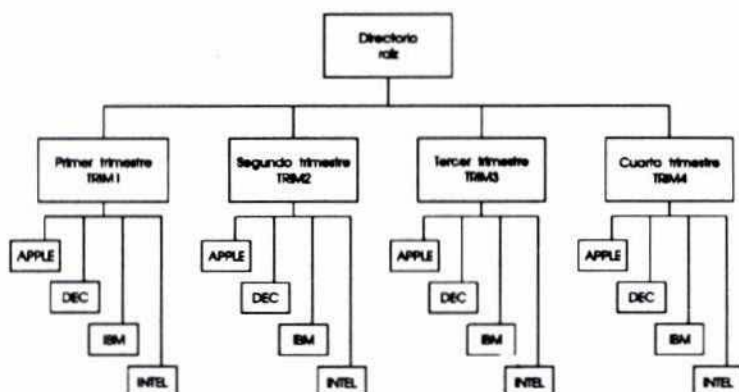
Estructura del directorio de archivo del agente de bolsa

El propósito de los subdirectorios es mejorar la organización del disco. Si, por ejemplo, el agente de bolsa desea estar al tanto de los precios de las acciones por trimestre del año fiscal, puede crear los subdirectorios que se muestran en la siguiente figura:



Subdirectorios para las cotizaciones de bolsa por trimestre

Puede separar aún más los archivos para compañías dentro de cada uno de los cuatro subdirectorios, como en la figura siguiente:



Distinción por compañías en subdirectorios

El uso de subdirectorios por parte del agente de bolsa hace que sus archivos de disco le resulten mucho más fáciles de encontrar y de mantener.

El DOS proporciona las cinco órdenes siguientes, específicamente para manipular directorios:

MKDIR
CHDIR
RMDIR
TREE
PATH

Antes de examinar las órdenes de subdirectorios del DOS, es importante entender cómo representa DOS el directorio raíz y los nombres de los subdirectorios.

Al dar formato a un disco en la unidad A y dar la orden:

DIR

DOS responde:

Volumen en unidad A sin etiqueta
Directorio de A:\

No se encontraron archivos

El DOS nos dice que el listado de directorio es de A:\. Evidentemente la unidad en curso es A. La barra invertida \ indica cuál es el directorio en curso. En este caso, \ nos dice que estamos en el directorio raíz. Como se ha indicado anteriormente, el directorio raíz proporciona al DOS el punto de arranque a partir del cual salen todos los demás subdirectorios. Si nuestro disco tiene la estructura que se muestra en la figura anterior, se alude a cada uno de los subdirectorios como sigue:

\APPLE
\DEC
\IBM
\INTEL

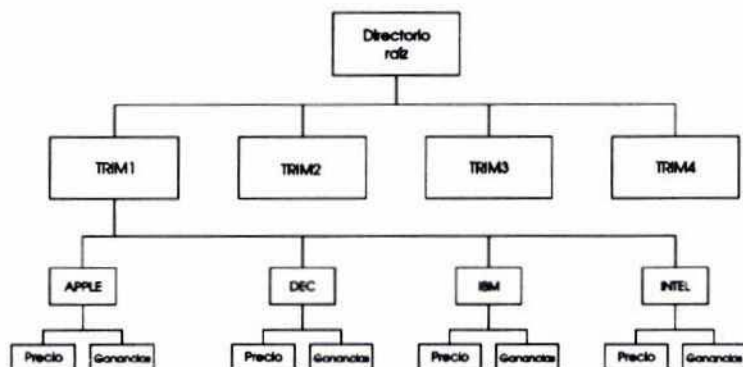
Si, por ejemplo, se examina el nombre \IBM, la barra invertida \ dice al DOS que empiece en el directorio raíz. IBM es el nombre de un subdirectorio contenido en el directorio raíz.

Considere la estructura de directorios de disco mostrada. Para hacer alusión a los archivos contenidos en el primer trimestre, se empieza en el directorio raíz y se baja un nivel, como sigue:

\TRIM1

Para hacer alusión al subdirectorio de IBM dentro de \TRIM1, se especifica el camino:

\TRIM1\IBM



Estructura de directorio con precios y ganancias

Una ruta o camino es simplemente la serie de nombres de subdirectorios que debe combinar el DOS para obtener el archivo que se desea. El DOS limita la longitud del camino a 63 caracteres. Si se nos da la estructura de directorio que se muestra en la figura anterior, lo que sigue son caminos válidos:

****especifica el directorio raíz

\TRIM1 alude al subdirectorio del primer trimestre

\TRIM3\IBM\GANANCIAS solicita el subdirectorio que contiene las ganancias de IBM durante el tercer trimestre (que no se muestra en la figura anterior).

Tamaño del disco	Número máximo de subdirectorios en el directorio raíz
160K	64
180K	
320K	112
360 y 720K	112
1.2M y 1.44M	224
Disco Rígido	Depende del tamaño del volumen

Subdirectorios permitidos por el directorio raíz.

El DOS utiliza el punto (.) como abreviatura del subdirectorio en curso. A la orden:

DIR .

El DOS responderá:

Volumen en unidad A sin etiqueta
Directorio de A:\

```
.          <DIR>      28/10/86      9:52a
..         <DIR>      28/10/86      9:52a
2 Archivo(s)      358400 bytes libres
```

El DOS utiliza *dos puntos* (..) para representar el directorio que está en el nivel superior al del subdirectorio en curso. Suponiendo que el directorio en curso sea \IBM, *punto* (.) señala al subdirectorio \IBM y *dos puntos* (..) señala al directorio raíz.

Con la orden:

DIR..

El DOS responderá:

Volumen en unidad A sin etiqueta

Directorio de A:\

IBM	<DIR>	28/10/86	9:52a
-----	-------	----------	-------

DEC	<DIR>	28/10/86	9:56a
-----	-------	----------	-------

INTEL	<DIR>	28/10/86	9:56a
-------	-------	----------	-------

APPLE	<DIR>	28/10/86	9:56a
-------	-------	----------	-------

4 Archivo(s)	3584400 bytes libres
--------------	----------------------

ANEXO No. 37

COMANDO MKDIR

MKDIR es un comando interno que crea nuevos directorios y subdirectorios. Un subdirectorio es un directorio que está dentro de otro directorio. El formato de la orden se presenta de dos formas:

`MKDIR [d:]ruta`

`MD [d:]ruta`

Donde:

d: es el nombre de la unidad de disco que contiene el disco al cual se desea crear el directorio.

d: es también el nombre del directorio raíz del disco en esa misma unidad de disco. Todos los directorios que se crean, son sub-directorios del directorio raíz.

ruta es el nombre de paso o camino que se debe especificar para tener acceso al nuevo directorio. Por ejemplo, el comando

MKDIR \PROGRAMS \OLDSTUFF

crearía un nuevo directorio con el nombre *OLDSTUFF* bajo el directorio existente *PROGRAMS*.

» Note como la ruta tiene rayas diagonales o **backslash** (\) para separar el directorio existente del nuevo.



ANEXO No. 38

COMANDO CHDIR

CHDIR es un comando interno que cambia de directorio o muestra el directorio corriente. ¿Qué quiere decir *directorio corriente*? Es el directorio por defecto donde DOS busca para encontrar los archivos. Cualquier archivo que se especifique sin una ruta explícita, debe existir en el directorio corriente para tener acceso a él.

El formato de la orden es:

```
CHDIR [[d:]ruta]
```

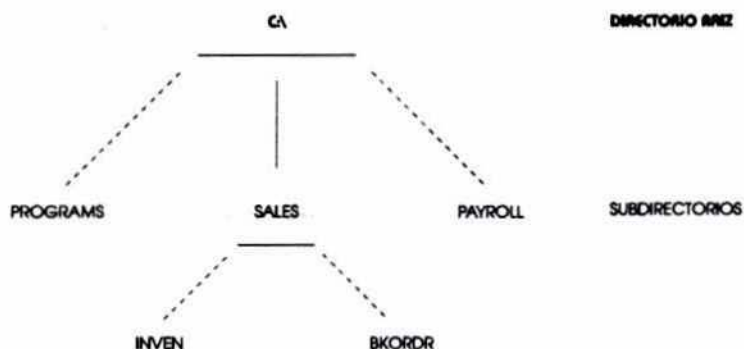
también

```
CD [[d:]ruta]
```

Para cada unidad de disco del sistema se puede tener un directorio corriente asignado. DOS recuerda el directorio corriente y si no se especifica una ruta de directorio, cualquier referencia a una unidad de discos servirá para tener acceso al directorio corriente.

Si sólo se digita **CD** se mostrará el directorio corriente de la unidad de disco por defecto.

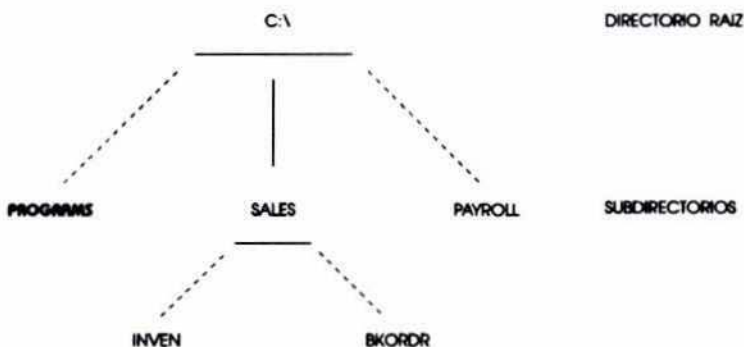
Se puede tener acceso a directorios de muchos niveles de profundidad. A continuación veremos algunos ejemplos.



En la estructura de disco del ejemplo, se tienen los subdirectorios *PROGRAMS*, *SALES*, *PAYROLL*, *INVEN*, y *BKORDR*. Para cambiar del directorio corriente C:, al directorio *PROGRAMS*, se debe digitar:

CD PROGRAMS

PROGRAMS queda ahora asignado como directorio corriente.

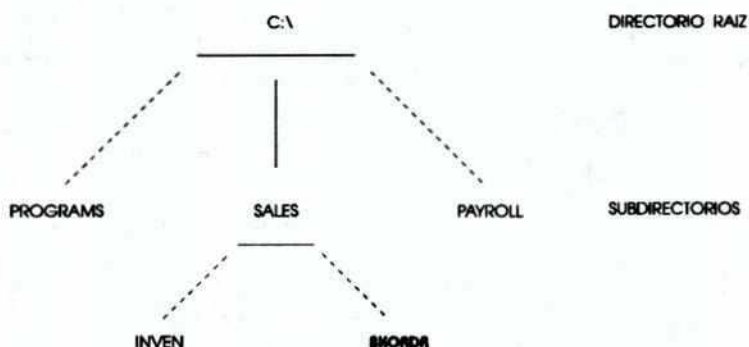


CD C:\SALES\INVEN

El directorio corriente es *PROGRAMS*. Este comando lo cambia a *INVEN*.

CD C:\SALES\BKORDR

El directorio corriente será *BKORDR* después de ejecutar este comando.



CD ..\INVEN

Este ejemplo muestra un método rápido de cambiar el directorio corriente de *BKORDR* a *INVEN*.

Los dos puntos (..), le dicen a DOS que suba un nivel en el árbol del directorio.

CD \

Esto cambia el directorio corriente al directorio raíz. El directorio raíz es el mismo que el de la unidad de discos. En este caso *C:*.



ANEXO No. 39

COMANDO RMDIR

RMDIR es un comando interno, usado para remover un directorio. El directorio debe estar vacío. No puede contener subdirectorios o archivos en él. El directorio raíz y el corriente, no pueden ser removidos. Su formato es:

RMDIR [d:]ruta

también

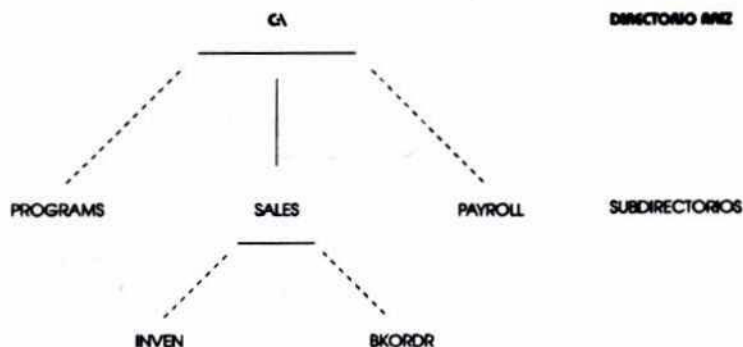
RD [d:]ruta

Donde:

[d:] es el nombre de la unidad de disco duro o disquete que tiene el directorio que se desea eliminar.

ruta es el nombre del directorio que se desea eliminar. Si la ruta define un subdirectorio, se debe especificar también el nombre del directorio al cual pertenecen, a menos que éste sea el directorio corriente.

Ejemplo:



En la estructura del disco del ejemplo están los subdirectorios *PROGRAMS*, *SALES*, *PAYROLL*, *INVEN*, y *BKORDR*.

Para remover el directorio *INVEN* con el comando **RD**, solamente se digita

RD C:\SALES\INVEN

y se pulsa la tecla **ENTER**

ANEXO No. 40

COMANDO TREE

La orden **TREE** del DOS nos permite ver la estructura del directorio de un disco. La orden muestra todos los subdirectorios (y opcionalmente sus archivos) en la pantalla.

El formato de la orden es:

```
[unidad:][camino]TREE [unidad_origen:][ /F]
```

Donde ocurre lo siguiente:

unidad: es la unidad de disco que contiene el archivo *TREE.COM*. Si no se especifica una unidad, el DOS utiliza la unidad de disco por defecto en curso.

ruta es el nombre del subdirectorio que contiene la estructura de directorio que hay que ver en la pantalla. Si no se especifica una unidad de origen, **TREE.COM** utiliza la unidad de disco por defecto en curso.

/F es un modificador que indica a *TREE* que muestre los nombres de cada uno de los archivos contenidos en los subdirectorios. Si no se especifica **/F**, **TREE** mostrará solamente los nombres de los subdirectorios.



Con la orden siguiente:

TREE

el sistema mostrará todos los subdirectorios contenidos en la unidad de disco por defecto en curso. Si se tiene un disco del DOS en la unidad A y un disco con los subdirectorios \APPLE, \DEC, \IBM e \INTEL en la unidad B, a la orden:

TREE B:

el sistema responderá mostrando:

LISTADO DE VIAS DE ACCESO AL DIRECTORIO

Vía de acceso:	\IBM
Sub-directorios:	TRIM1
Vía de acceso:	\IBM\TRIM1
Sub-directorios:	Ninguno
Vía de acceso:	\DEC
Sub-directorios:	Ninguno
Vía de acceso;	\INTEL
Sub-directorios:	Ninguno
Vía de acceso:	\APPLE
Sub-directorios:	Ninguno

Si se transcribe la orden:

TREE B:/F

Esto mostrará los nombres de los archivos contenidos en cada uno de los subdirectorios de la unidad B.

ANEXO No. 41

COMANDO PATH

La orden **PATH** del DOS proporciona un método conveniente de decirle al DOS dónde debe buscar las órdenes externas. Cada vez que se ejecuta una *orden externa* (ésto es, un archivo del DOS con una extensión *EXE* o *COM*), el DOS busca primero el archivo de la orden en el directorio por defecto en curso. Si el archivo existe, el DOS ejecuta la orden. En caso contrario, el DOS hace comprobaciones para ver si hemos definido un camino hacia otros subdirectorios o unidades de disco en qué buscar la orden.

El formato de la orden **PATH** es:

PATH [unidad:][;[unidad:][camino]...]

A partir de la siguiente orden:

PATH B:

si el DOS no consigue localizar una orden externa dada en el subdirectorio en curso, buscará el archivo de la orden en el directorio por defecto de la unidad **B**.

PATH \IBM; \DEC; \INTEL; \APPLE

el DOS buscará primero el archivo de orden externo en el directorio por defecto y después buscará el archivo de la orden secuencialmente por los caminos que se han proporcionado a

través de la orden **PATH**. La Figura que aparece en este Anexo es un diagrama de flujo que describe este proceso.

Si el DOS no localiza el archivo en el directorio actual, comenzará la búsqueda del archivo en el subdirectorio **\IBM**. El DOS continuará su búsqueda secuencialmente por los caminos especificados, hasta que se encuentre el archivo o hasta que se agote el camino. Si el DOS no encuentra el archivo de la orden, mostrará el siguiente mensaje de error:

Mandato o nombre de archivo erróneo

Si el camino especificado en la orden **PATH** contiene una entrada incorrecta, el DOS prescindirá de esa entrada y continuará con el camino siguiente. Es importante observar que la orden **PATH** no comprueba la validez de las entradas en el momento en que se da. En lugar de hacer ésto, el DOS mostrará el mensaje de error cada vez que intente acceder al camino incorrecto. La orden:

PATH;

borra el camino anterior e indica al DOS que examine solamente el directorio por defecto en curso al buscar órdenes. Si se da la orden **PATH** sin ningún parámetro, el DOS mostrará el camino en curso como sigue:

PATH

PATH=\IBM;\DEC;\INTEL;\APPLE;

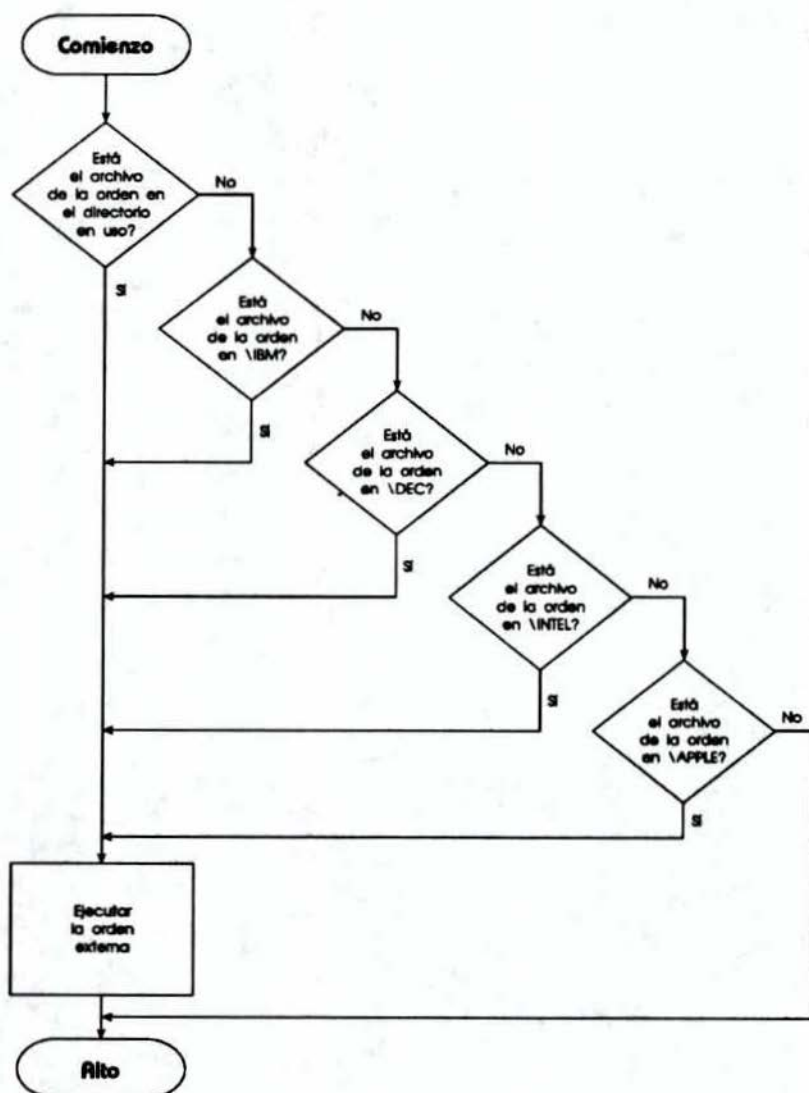


Diagrama que muestra la ejecución secuencial de PATH



ANEXO No. 42

PROCESAMIENTO DE LOTES

Normalmente, cada vez que damos una orden esperamos a que termine de ejecutarse antes de introducir una segunda. Estamos en constante interacción con el computador. El procesamiento de este tipo se denomina **procesamiento interactivo**. El usuario digita una orden desde el indicador del sistema y espera entonces a que la orden concluya antes de seguir adelante. Sinembargo, existen varias ocasiones en las cuales el procesamiento no exige una interacción constante en el usuario. Consideremos una aplicación que calcula los sueldos de unos empleados, ordena los sueldos e imprime los talones. Si se supone que las órdenes **NOMINA.EXE**, **ORDDATOS.EXE** e **IMPTALON.EXE** llevan a cabo los pasos de procesamiento, se podrían dar todas las órdenes interactivamente. En este caso, se da primero la orden **NOMINA.EXE**.

Cuando el procesamiento ha terminado, se clasifican los sueldos con la orden **IMPTALON.EXE**, como se muestra aquí:

NOMINA
ORDDATOS
IMPTALON.EXE

En este momento comienza la impresión de talones, y podemos ir a trabajar en alguna otra tarea, lejos del computador. Como hemos llevado a cabo todas las órdenes de la nómina interactivamente, hemos tenido que permanecer junto al computador para dar la orden siguiente a medida que termina cada proceso. Dependiendo del tamaño de la compañía, la cantidad de tiempo que se esperaría para dar la orden siguiente podría ser considerable.

En este caso, ninguna de las órdenes requiere interacción con el usuario, una vez ha empezado el programa. En tales casos, el DOS nos permite agrupar todas las órdenes en un archivo que él lee y ejecuta de manera secuencial. Consideremos el ejemplo de la nómina anterior. El DOS nos permite agrupar las tres órdenes en un archivo con la extensión *BAT*, como se ilustra aquí:

**NOMINA
ORDDATOS
IMPTALON**

Si se ponen las tres órdenes en el archivo *HACERNOM.BAT*, entonces se da simplemente la orden **HACERNOM** desde el indicador del DOS. El DOS leerá y ejecutará todas las órdenes del archivo secuencialmente. Ahora queda una libre para llevar a cabo otras tareas lejos del computador, mientras el DOS ejecuta el archivo por lotes. El procesamiento en el cual el sistema obtiene las órdenes a partir de un archivo, en lugar de obtenerlas del usuario a través del teclado, se denomina **procesamiento por lotes**.

- » Obsérvese que la extensión *BAT* identifica un archivo con órdenes para procesamiento por lotes.

Por defecto, el DOS muestra el nombre de cada programa en la pantalla, a medida que ejecuta. La ejecución del archivo *HACERNOM.BAT* da lugar a la pantalla siguiente:

```
A>NOMINA
A>ORDDATOS
A>IMPTALON
```

Al agrupar las órdenes de la nómina en un solo archivo, *BAT* ahorra una cantidad significativa de tiempo que en otro caso se desperdiciaría esperando a que terminase cada programa para poder dar la orden siguiente. Además, hay algunas aplicaciones que están formadas por varios programas pequeños, que se ejecutan secuencialmente. En estos casos, el procesamiento por lotes no sólo ahorra tiempo, sino que además ahorra mucha escritura.

Suponga que desea llevar a cabo lo siguiente:

Borrar la pantalla.
 Pedir la fecha al usuario.
 Pedir la hora al usuario.
 Ver la versión del DOS en curso.

Se puede crear un archivo llamado *PONFECHA.BAT* el cual debería tener un aspecto como el siguiente:

CLS
DATE
TIME
VER

Al ejecutar el archivo *PONFECHA.BAT* ocurre lo siguiente:

A>DATE

La fecha actual es Jue 2/10/86

Introduzca nueva fecha: (dd/mm/aa):

A>TIEMPO

La hora actual es 11:31:23,03

Introduzca nueva hora:

A>VER

IBM Personal Computer - Versión DOS 3.00

Las siguientes reglas proporcionan las líneas maestras para crear procedimientos por lotes:

- No dé nombres de órdenes internas del DOS tales como *CLS.BAT* a los archivos por lotes. El DOS no podrá encontrar el archivo *BAT*.

- El DOS ejecuta los archivos *COM* y *EXE* antes de ejecutar archivos *BAT* con el mismo nombre, con cargo a la estrategia de búsqueda indicada en la orden *PATH*.
- Los archivos por lotes deben tener la extensión *BAT*.
- Si se invoca un proceso por lotes que reside en un disco flexible y después se saca el disco durante el procesamiento, el DOS le pedirá que vuelva a insertar el disco al terminar el programa en curso.
- Al cambiar la unidad de disco o el directorio por defecto durante la ejecución de un archivo *BAT* no resultará afectando el procedimiento por lotes.
- Se puede invocar a otros archivos tipo *BAT* desde el interior de un proceso por lotes, si el segundo procedimiento por lotes es la última orden del archivo; por ejemplo, si se quiere invocar *HACERNOM.BAT*, se alude simplemente a su nombre como sigue:

```
CLS  
DATE  
TIME  
HACERNOM
```

- Si el segundo procedimiento por lotes aparece en medio del archivo por lotes, hay que invocar un procesador de órdenes secundario para posibilitar que el control regrese al programa principal:

```
CLS  
DATE  
TIME  
COMMAND/C HACERNOM  
CLS
```

La sentencia **COMMAND/C** carga un procesador de órdenes secundario para ejecutar el procesamiento por lotes.

NOTA:

A partir de la versión 3.30, la línea 4 puede reemplazarse por **CALL HACERNOM**

- Cualquier orden que se pueda dar desde el indicador del DOS puede darse desde el interior de un archivo por lotes.

CREACION DE ARCHIVOS PARA PROCESO DE LOTES

Una de las formas más sencillas y rápidas de crear un archivo por lotes, es sencillamente copiar la serie de órdenes desde el teclado como sigue:

COPY CON NOMBARCH.BAT

Cuando haya introducido la última línea del archivo por lotes, pulse la tecla **F6** y pulse **ENTER**. El anterior ejemplo creaba un archivo por lotes que contenía:

```
CLS
DATE
TIME
VER
```

Para ingresar el archivo desde el teclado, límitese a transcribir:

COPY PONFECHA.BAT

Al terminar el proceso la pantalla contendrá:

```
A>COPY CON PONFECHA.BAT
CLS
DATE
TIME
VER
^Z
```

```
1 Archivo(s) copiado(s)
```

Hay muchos usuarios del DOS que invocan a un procesador de textos para crear sus archivos por lotes, o al editor de líneas de DOS EDLIN.

El DOS proporciona varias órdenes que mejoran el procesamiento por lotes. Algunas órdenes son:

REM [mensaje]
PAUSE [mensaje]
ECHO [on/off/mensaje]
GOTO [etiqueta]
IF [NOT] condición Orden_del_DOS
FOR %% variable **IN** (conjunto) **DO** Orde_del_DOS
SHIFT

REM

La orden para lotes **REM** nos permite ver mensajes en la pantalla durante el procesamiento por lotes. El formato de orden es:

REM [mensaje]

Donde

Mensaje

es una cadena de carácter opcional que contiene hasta 123 caracteres

Obsérvese que cuando se introducen las siguientes órdenes **REM** dentro de *HACERNOM.BAT*, los mensajes mostrados en la pantalla pueden mejorarse:

REM Estoy empezando con NOMINA
NOMINA
REM Estoy ordenando los datos de NOMINA

ORDDATOS

REM Estoy imprimiendo los talones de la NOMINA
IMPTALON

Ahora la ejecución de *HACERNOM.BAT* da lugar a:

```
A>REM Estoy empezando con NOMINA
A>NOMINA
A>REM Estoy ordenando los datos de NOMINA
A>ORDDATOS
A>REM Estoy imprimiendo los talones de la NOMINA
A>IMPTALON
```

Como se verá, la orden **ECHO OFF** (desactivado), que se presenta más adelante en este anexo, suprime la visualización, de mensajes a través de **REM** o **PAUSE**.

PAUSE

La orden para lotes **PAUSE** es similar a **REM** en cuanto nos permite ver mensajes dentro de un archivo por lotes. Además de mostrar mensajes, **PAUSE** suspende temporalmente el procesamiento hasta que nos damos por enterados del mensaje, pulsando cualquier tecla para continuar. Si no se desea continuar, se pulsa **CTRL-BREAK** o **CTRL-C** dando lugar a que DOS responda:

```
¿Terminar el trabajo por lotes (S/N)?
```

Basta con pulsar **S** para finalizar el procesamiento por lotes, o **N** para continuar.

Si se supone que el procedimiento *HACERNOM.BAT* requiere un disco en blanco en la unidad **B** antes de la clasificación, introduzca la orden **PAUSE** como sigue:

NOMINA

PAUSE Coloque un disco en blanco en la unidad **B**:

ORDDATOS

IMPTALON

La ejecución de *HACERNOM.BAT* da lugar a:

A>NOMINA

A>PAUSE Ponga un disco en blanco en la unidad **B**:
Pulse cualquier tecla para continuar...

A> ORDDATOS

A>IMPTALON

ECHO

La orden por lotes **ECHO** permite o impide ver los nombres de órdenes del DOS cuando se ejecutan desde el interior de un archivo *BAT*. Cuando está desactivada (*OFF*), **ECHO** inhibe la visualización de nombres de órdenes y mensaje de **REM**. **ECHO** no inhibe la salida producida por las órdenes.

El formato **ECHO** es:

ECHO [ON/OFF/mensaje]

Donde:

Mensaje es un mensaje que debe visualizar **ECHO**.

La longitud de este mensaje no debe sobrepasar los 17 caracteres (en DOS versión 3.0). Cuando **ECHO** está activado, el DOS muestra los nombres de todas las órdenes en la pantalla, a medida que las ejecuta; si no se especifica ningún parámetro a **ECHO**, el estado actual (*ON* u *OFF*, activado o desactivado) se ve en la pantalla como se muestra aquí:

ECHO

```
ECHO es ON
```

GOTO

La orden por lote **GOTO** proporciona un mecanismo para hacer un salto desde dentro de un archivo por lotes. El formato de **GOTO** es:

```
GOTO [etiqueta]
```

Donde:

Etiqueta es el nombre de una etiqueta dentro del archivo por lotes en la cual el DOS puede proseguir la ejecución.

Las etiquetas pueden tener prácticamente cualquier longitud. Sin embargo, sólo son significativos los ocho primeros caracteres de la etiqueta. El DOS, por tanto, considera que las etiquetas **:nombreetiqueta1** y **:nombreetiqueta2** son equivalentes. Al

introducir el siguiente procedimiento *BAT* se verá repetidamente un listado del directorio, hasta que el usuario pulse **CTRL-BREAK** para finalizar la ejecución.

```
:bucle  
DIR  
GOTO bucle
```

IF

El DOS proporciona la orden **IF** para admitir el procesamiento condicional dentro de las órdenes por lotes. El formato de **IF** es:

IF [NOT] condición Orden_del_DOS

Donde:

Condición

es la condición lógica que evalúa el DOS.

La condición puede ser una de las siguientes:

- **ERRORLEVEL** *número*
- **EXIST** *especificación_archivo*
- **cadena1 = cadena2**

ERRORLEVEL *número* se evalúa como *TRUE* (verdadero), cuando el programa anterior sale al DOS con un estado de error mayor o igual que el valor decimal contenido en *número*. Una de las posibilidades que el DOS ofrece a los programas es la capacidad de proporcionar un valor de estado al DOS cuando concluye el programa.

```
NOMINA
ORDDATOS
IF ERROR LEVEL 1 GOTO IMPRIMIR
REM HAY UN ERROR EN LA CLASIFICACION
GOTO HECHO
:IMPRIMIR
IMPTALON
:ECHO
```

En el ejemplo de archivo **BAT**, aparece el siguiente mensaje de error:

```
REM HAY UN ERROR EN LA CLASIFICACION
```

El mensaje se ve en la pantalla si el programa *ORDDATOS* sale al DOS con un estado menor que 1. En caso contrario, el procedimiento prosigue simplemente hasta su conclusión.

EXIST *especificación_de_archivo* se evalúa como *TRUE* (*verdadero*), cuando el archivo nombrado existe como se ha especificado.

Considere el contenido del siguiente archivo por lotes:

```
NOMINA
IF EXIST NOMINA.SRT GOTO IMPRIMIR
ORDDATOS
:IMPRIMIR
:IMPTALON
```

Si los datos de la nómina ya han sido clasificados (*NOMINA.SRT*), este archivo *BAT* imprime los talones. En caso contrario, el procedimiento invoca al programa *ORDDATOS* e imprime entonces los talones.

La condición **cadena1 = cadena2** se evalúa como *TRUE* (*verdadero*) cuando la cadena de caracteres contenida en **cadena1** es idéntica a la que hay en **cadena2**. La comparación tiene en cuenta las mayúsculas y minúsculas, así como las cadenas tienen que coincidir exactamente. Si se digita lo siguiente, el DOS evaluará como *FALSE* (*falso*) la condición:

IF PRIMERA CADENA = primera cadena GOTO HECHO

La capacidad de llevar a cabo comparaciones de cadenas pasará a ser más vital cuando se traten los parámetros por lotes, más adelante en este anexo.

El operador **NOT** invierte simplemente el resultado de la condición puesta en práctica. Considere la orden siguiente:

IF NOT EXIST NOMINA.SRT ORDDATOS

FOR

La orden **FOR** proporciona un mecanismo para hacer procesamiento repetitivo desde dentro de un archivo *BAT*. El formato de **FOR** es:

FOR %%variable IN [conjunto] DO orden_del_DOS

Donde ocurre lo siguiente:

- %%variable** es una variable a la que el DOS da secuencialmente los valores del conjunto dado.
- Conjunto** es un conjunto de nombres de archivo que el DOS debe usar secuencialmente.

Orden_del_DOS es la orden que el DOS lleva a cada interacción.

Considere el siguiente archivo *BAT*:

```
FOR %%F IN (AUTOEXEC.BAT CONFIG.SYS PRUEBA.BAT)  
DO TYPE %%F
```

El doble signo de tanto por ciento que hay **%%F** indica al DOS que **F** es una variable cuyo valor puede ser cambiado por el DOS durante la ejecución del archivo por lotes. El DOS asigna primero a la variable **F** el valor *AUTOEXEC.BAT*, y escribe su contenido. Acto seguido, el DOS asigna a **F** el valor de *CONFIG.SYS* y lo muestra. Finalmente, el DOS asigna a **F** el valor *PRUEBA.BAT* y lo escribe y el procedimiento por lotes concluye. Como se verá, los caracteres ambiguos son aceptables como elementos de conjunto.

Si introduce el ejemplo siguiente, se escribirán todos los archivos *BAT* que haya en el directorio en curso:

```
FOR %%F IN (*.BAT) DO TYPE %%F
```

Si un miembro del conjunto no es aceptable para la orden del DOS, el DOS se limita a seguir el bucle **FOR** con el miembro siguiente.

Una de las características más potentes que proporciona el DOS para procesamiento por lotes es la capacidad de pasar parámetros (valores) a procedimiento por lotes. Considere la siguiente línea de órdenes:

```
ARCHIBAT PARAM1 PARAM2 PARAM3
```

En este caso, **ARCHIBAT** es el archivo *BAT* que va a ejecutar el DOS. **PARAM1**, **PARAM2** y **PARAM3** son los parámetros o valores que se van a pasar al procedimiento por lotes. El DOS accede secuencialmente a los parámetros dentro de procedimientos por lotes.

En el ejemplo anterior el DOS asigna los parámetros como sigue:

%1 contiene el primer parámetro, **PARAM1**.
%2 contiene el segundo parámetro, **PARAM2**.
%3 contiene el tercer parámetro, **PARAM3**.

El DOS admite diez parámetros por lote (de 0% a %9). Hemos visto anteriormente el signo doble de tanto por ciento que se usa para denotar una variable dentro de un procedimiento por lotes. En este caso, el signo simple de tanto por ciento seguido por un único dígito, denota un parámetro o valor que está pasando el usuario al procedimiento por lotes. Al introducir el siguiente procedimiento por lotes, se verán los parámetros del 1 al 9. Si un parámetro no está presente, **ECHO** se limita a ignorar la referencia hecha a él:

ECHO %1 %2 %3 %4 %5 %6 %7 %8 %9

El parámetro 0, (**%0**) es único en cuanto contiene el nombre del procedimiento por lotes. Considere un archivo por lotes llamado **PONHORA.BAT** que contiene lo siguiente:

ECHO %0
DATE
TIME

Al invocar a *PONHORA* se obtiene:

PONHORA

```
A>ECHO PONHORA
La fecha actual es Jue 2/10/86
Introduzca nueva fecha: (dd/mm/aa):

A>TIME
La hora actual es 12:02:22,22
Introduzca nueva hora:

A>
```

Hay muchas aplicaciones que exigen que se compruebe si el usuario ha proporcionado un cierto parámetro. El DOS asigna el valor NULL a todo parámetro al que el usuario no haga referencia. Para detectar una cadena NULL, ponga el parámetro entre comillas simples, como sigue:

IF '%1' = " GOTO NOHAYPARAM

Si el parámetro %1 no existe, la situación del DOS da lugar a:

```
IF "==" THEN GOTO NOHAYPARAM
```

La orden **REN** (cambiar de nombre) del DOS no nos permite cambiar el nombre de un archivo entre unidades distintas. Si se intenta una acción como ésta, sucederá lo siguiente:

REN CONFIG.SYS B:CONFIG.SYS

```
Parámetro erróneo
```

El siguiente archivo por lotes, *MUEVE.BAT*, mueve un archivo entre unidades asegurándose primero de que el usuario haya proporcionado los nombres del original y del destino. Si están presentes ambos nombres, el procedimiento comprueba si existe ya un archivo con el mismo nombre que el de destino. De ser así, el procedimiento nos indica:

PAUSE EL ARCHIVO DEL DESTINO YA EXISTENTE
Pulse cualquier tecla para continuar...

Para continuar con el procedimiento de cambio de nombre, límtese a pulsar una tecla. Si no desea seguir seguir adelante, pulse **CTRL-BREAK**. Si continúa el procesamiento, *MUEVE* copia el archivo original en el destino especificado. *MUEVE* pide entonces al usuario que pulse una tecla antes de borrar el archivo, como se muestra en lo que sigue:

```
IF '%1'=='' GOTO NOHAYPARAM
IF '%2'=='' GOTO NOHAYPARAM
IF EXIST %2 PAUSE EL ARCHIVO DE DESTINO YA EXISTE
COPY %1 %2
PAUSE ESTOY A PUNTO DE BORRAR EL ARCHIVO ORIGINAL
DEL %1
GOTO HECHO
:NOHAYPARAM
REM ES NECESARIO ESPECIFICAR LOS ARCHIVOS FUENTE Y DESTINO
:HECHO
```

Al introducir el siguiente procedimiento *BAT* se escribirán en pantalla todos los archivos del conjunto especificados por %1:

```
FOR %%F IN (%1) DO TYPE %%F
```

Si se da el nombre *T.BAT* al archivo, se puede introducir:

```
T *.*
```

La orden muestra el contenido de todos los archivos del directorio en curso.

Digite la orden:

T *.BAT

Se verán los contenidos de todos los archivos *BAT* del directorio en curso. Digite:

T

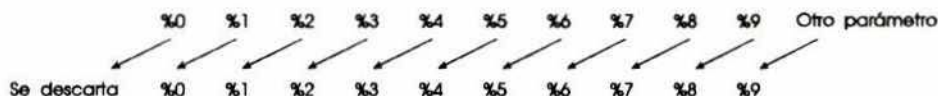
No se ve archivo alguno.

SHIFT

Si la aplicación requiere más de diez parámetros para procesamiento por lotes, se puede utilizar la orden **SHIFT**. El formato es:

SHIFT

SHIFT se limita a hacer rotar los parámetros un lugar hacia la izquierda. En otras palabras, se asigna a %0 el valor de %1, a %1 el valor de %2 y así sucesivamente. Este concepto está ilustrado en la siguiente figura:



Ejemplo de cómo SHIFT desplaza todos los parámetros una posición hacia la izquierda.

Si existen más de diez parámetros, el DOS asigna a %9 el valor del siguiente parámetro en la lista. Si introduce el siguiente procedimiento *MOSTRAR.BAT*, éste mostrará todos los parámetros que reciba:

```
:BUCLE  
SHIFT  
IF '%0' = " GOTO HECHO  
ECHO %0  
GOTO BUCLE  
:HECHO
```

El procedimiento se invoca en la forma:

MOSTRAR 1 2 3 4 5 6 7 8 9 10 11

AUTOEXEC.BAT

Cada vez que se arranca el sistema, el DOS examina el directorio raíz en busca de un archivo llamado *AUTOEXEC.BAT*. Si se encuentra el archivo, el DOS ejecuta todas las órdenes que contenga. Si no encuentra el archivo, el DOS emite las órdenes **DATE** y **TIME**.

El *AUTOEXEC.BAT* no se distingue en nada de otros procedimientos **BAT**. Hay que seguir todas las reglas existentes para crear archivos por lotes, y son válidas todas las órdenes por lotes.

Considere el siguiente procedimiento *AUTOEXEC.BAT* que pide al usuario la fecha y la hora, borra la pantalla y entonces muestra la versión del DOS en curso.

```
DATE  
TIME  
CLS
```

Si está desarrollando aplicaciones para usuarios que deseen ejecutar solamente nuestro sistema particular, límtese a poner la referencia a su programa dentro de **AUTOEXEC.BAT**. El usuario nunca tendrá que preocuparse del indicador del DOS.

Aunque el procesamiento por lotes es cómodo, es importante llevar a cabo tanta cantidad del procesamiento como sea posible, desde un lenguaje de alto nivel. Los archivos *BAT* se ejecutan lentamente porque el DOS tiene que interpretar cada orden antes de ejecutarla. Así, los archivos *BAT* no están hechos para usarlos como lenguaje de programación.

ANEXO No. 43

ADECUACION DEL DOS

Este anexo examina las formas en que se puede modificar el DOS para adecuarlo a sus necesidades específicas. Hay muchos usuarios del DOS que encuentran que pueden mejorar el rendimiento y capacidad del DOS, modificando uno o más valores de configuración de los que utiliza el DOS cada vez que arranca el sistema. Sin embargo, si no se lleva a cabo una gran cantidad de programación, ni se utilizan grandes aplicaciones de bases de datos, los valores por defecto que proporciona el DOS para configurar el sistema operativo, deberían ser correctas.

Cada vez que arranca, el DOS se carga en memoria con espacio reservado para órdenes internas, estructuras de datos del DOS y controladores de dispositivos y otros programas residentes en memoria.



Utilización de la memoria del DOS

Las órdenes internas del DOS se han examinado en los anexos anteriores. Las estructuras de datos del DOS son posiciones de almacenamiento que utiliza el DOS para llevar la cuenta de cosas tales como archivos abiertos, buffers (memorias intermedias) de transferencia a disco e información específica del país para las órdenes internacionales del DOS. Como se verá, los controladores de dispositivos son programas que permiten al DOS comunicarse con dispositivos físicos tales como la impresora, el teclado y la pantalla.

El DOS nos permite modificar varias de sus características cada vez que vuelve a arrancar el sistema. Para hacer ésto, se modifican uno o más de los parámetros de configuración que hay en el archivo *CONFIG.SYS*, y que debe residir en el directorio raíz.

Algunos de los parámetros que el DOS nos permite modificar son:

BREAK
BUFFERS
COUNTRY
DEVICE
DRIVPARM
FCBS
FILES
LASTDRIVE
SHELL

Cada vez que arranca el sistema, el DOS busca en el directorio raíz el archivo *CONFIG.SYS*. Si el archivo existe, el DOS lo lee secuencialmente y se configura a sí mismo en memoria con los valores contenidos en el archivo. Si el archivo no existe, el DOS utiliza valores por defecto para todos los parámetros enumerados.

Para crear el archivo *CONFIG.SYS* la mayoría de los usuarios del DOS utiliza el editor de textos **EDLIN** o copia el archivo *CONFIG.SYS* desde el teclado como sigue:

COPY CON CONFIG.SYS

```
...
< líneas de archivo >
...
^Z ↵
```

BREAK

(Por defecto: **BREAK=OFF**). En muchas ocasiones es conveniente detener un programa de aplicación pulsando **CTRL-C**. Por defecto, el DOS investiga si el usuario ha pulsado o no la combinación **CTRL-C** cada vez que escribe en la pantalla o en la impresora, o cuando lee el teclado. Es posible aumentar el número de funciones tras cuya conclusión va a examinar el DOS en busca de **CTRL-C**. Si se digita:

BREAK=ON

en el archivo *CONFIG.SYS*, el DOS buscará el **CTRL-C** cuando termine las lecturas y escrituras en disco. Esto aumenta la probabilidad de que el DOS intercepte el **CTRL-C** rápidamente y termine el programa según lo desee.

Al establecer **BREAK=ON** se añade una penalización a los tiempos. Como ahora el DOS tiene que comprobar si hubo un **CTRL-C** después de muchas operaciones, el procesamiento global será más lento. Para eliminar la comprobación de **CTRL-C** por parte del DOS, realice simplemente la orden siguiente en *CONFIG.SYS*:

BREAK=OFF

A continuación, vuelva a arrancar. El DOS seguirá comprobando si hay un **CTRL-C** cuando escriba en la pantalla o cuando lea el teclado.

BUFFERS

(Por defecto: en el **IBM PC: BUFFERS=2**; y en el **IBM PC AT BUFFERS=3**). La entrada **BUFFERS** de *CONFIG.SYS* permite definir el número de Buffers o segmentos de memoria intermedia de disco que utiliza el DOS para almacenar datos cuando se lee o escribe en el disco.

Un **Buffer** es una posición de almacenamiento en memoria de 528 bytes. Cada vez que la aplicación solicita del DOS que escriba en el disco, el DOS pone la información en un buffer del disco. Una vez que el buffer se llena, el DOS escribe (o vuelca) el contenido del Buffer en el disco.

Cuando el DOS lleva a cabo operaciones de lectura del disco, investiga primero en la lista de buffers para ver si el sector deseado está ya en memoria. En caso afirmativo, no se necesita *E/S* a disco. En caso contrario, el DOS lee el sector del disco y lo pone en un **Buffer**.

Si se incrementa el número de Buffers de disco a los que puede acceder el DOS, un programa que lleve a cabo *E/S* a disco con acceso aleatorio (un programa de base de datos), tiene más probabilidades de mejorar su velocidad al encontrar ya en memoria el registro deseado. Esto elimina muchas operaciones de *E/S* a disco.

Las aplicaciones que leen archivos secuencialmente no verán una mejora de rendimiento utilizando muchos buffers.

Lo que sigue son líneas generales para determinar el número correcto de buffers de disco:

- Las aplicaciones de bases de datos utilizan entre 10 y 25 buffers.
- Las aplicaciones de proceso de texto utilizan entre 5 y 20 buffers.
- Los grandes números de subdirectorios utilizan entre 10 y 25 buffers.
- Los sistemas de disco rígido utilizan tres buffers como mínimo.

COUNTRY

Se utiliza para especificar los siguientes parámetros para un país dado:

Formato de fecha **mm dd yy**

Símbolo monetario **\$**

Separador de millares **,**

Separador de decimales **.**

Separador de fechas **-**

Separador de horas **:**

Cifras decimales significativas **2**

Formato monetario **0**

Formato horario de 12 horas **(1-12)**

Separador de datos en listas **,**

Los países se especifican como se muestra en los manuales del DOS de su equipo.

Para seleccionar un país, especifique el código del país en *CONFIG.SYS* como sigue:

`COUNTRY=código_del_país`

Para seleccionar Italia, por ejemplo, coloque la entrada siguiente en *CONFIG.SYS*:

COUNTRY = 039

Si el valor del código del país no es válido, el DOS toma por defecto el formato de los Estados Unidos.

Remítase a la bibliografía recomendada para información adicional.

DEVICE

Cada dispositivo físico del sistema requiere un controlador de dispositivo, que es un programa que usa el DOS para comunicarse con él. Por defecto, el DOS proporciona controladores de dispositivo para entrada normal (el teclado), para la salida normal (la pantalla), para la impresora, para el reloj del sistema, y para unidades de disco rígido y de disco flexible. El DOS instala estos controladores de dispositivos en memoria cada vez que arranca el sistema. Los dispositivos que no son estándar en un PC, tales como un ratón, pueden necesitar un controlador de dispositivo especial. Normalmente, el fabricante suministra el controlador de dispositivo en un disco flexible.

Con el objeto de informar al DOS acerca del nuevo dispositivo, hay que instalar el controlador de dispositivo cada vez que arranca el sistema.

Para hacer ésto, coloque la entrada siguiente en *CONFIG.SYS*:

DEVICE=especificación_de_archivo

Supongamos que el controlador de dispositivo está contenido en el archivo *TRAZADOR.SYS*; la entrada *CONFIG.SYS* será:

DEVICE = TRAZADOR.SYS

El DOS suministra dos controladores de dispositivos que se pueden instalar, *ANSI.SYS* y *VDISK.SYS*. Se selecciona uno o el otro según nuestras necesidades. El primer controlador, *ANSI.SYS*, proporciona un mejor apoyo para el teclado y para la pantalla. Hay muchas aplicaciones que utilizan las capacidades de manipulación de pantalla que proporciona *ANSI.SYS* con el objeto de mantener la compatibilidad de la aplicación para diferentes computadores. Las capacidades de manipulación de pantalla de *ANSI* permiten a los programadores borrar la pantalla, borrar líneas de la pantalla y establecer los colores del primer plano y del fondo, así como situar el cursor. El controlador *ANSI* es un estándar de la industria para estas funciones. Esto significa que un programa que utilice las capacidades *ANSI* en un IBM PC correrá con éxito en un Rainbow de DEC, que utiliza un material de video completamente diferente. Para instalar el controlador *ANSI*, teclee la entrada siguiente en el archivo *CONFIG.SYS* y vuelva a arrancar:

DEVICE = ANSI.SYS

El controlador de dispositivo *VDISK.SYS* nos permite imitar con la memoria, a una unidad de disco. Una unidad de disco que reside en memoria de acceso aleatorio (**RAM**) se conoce con el nombre de disco en **RAM** o disco virtual. Se puede acceder rápidamente a los archivos contenidos en un disco en **RAM**, puesto que no hay dispositivos mecánicos que hagan lento el acceso. Los discos en **RAM** funcionan hasta diez veces más de prisa que los discos estándar.

En esencia, las unidades en RAM se tratan exactamente igual que se trataría a una unidad de disco rígido o flexible. La unidad en RAM pierde su contenido cuando se desconecta la corriente. Por tanto, toda la información que no se haya grabado desde el disco en RAM al disco flexible o al disco rígido se perderá al apagar el computador.

Las unidades en RAM tienen las características siguientes:

- El tamaño de la unidad lo especifica el usuario.
- Los discos virtuales tienen etiquetas de volumen y nombres de unidad. Si se dispone de un sistema con dos unidades de disco flexible **A** y **B**, al instalar una unidad en RAM se da lugar a la unidad **C**. El DOS nos permite instalar múltiples unidades en RAM.
- Cada unidad en RAM reserva aproximadamente 768 bytes además del número de bytes especificados para la unidad.
- El tamaño mínimo de una unidad RAM es 64K.
- Después de la instalación de la unidad en RAM, debe quedar sin usar un mínimo de 64Kb de RAM. En caso contrario, **VDISK** mostrará un mensaje de error y modificará el tamaño del disco en RAM para asegurar que esté disponible un mínimo de 64Kb.

El formato para instalar una unidad en RAM al arrancar el sistema es:

`DEVICE=VDISK.SYS Tamaño tamañosector numarchivos [/E]`

Donde ocurre lo siguiente:

Tamaño es el número de bytes que hay que reservar para la unidad en RAM.

Tamaño sector es el tamaño de cada sector en bytes. Los tamaños válidos son 128, 256 y 512 (el tamaño por defecto es de 128).

Numarchivos es el número de archivos que puede almacenar el disco virtual. El valor por defecto es 64. El valor debe estar en el intervalo entre 2 y 512. Cada entrada utiliza 32 bytes. *VDISK* utiliza una de las entradas para el nombre de volumen del disco.

/E se puede utilizar con un IBM PC AT para utilizar la extensión de memoria. La extensión de memoria se encuentra por encima del espacio de direcciones de 1Mb. El modificador **/E** obliga a *VDISK* a utilizar la extensión de memoria para el disco virtual.

Las que siguen son entradas válidas para *VDISK.SYS*:

DEVICE = VDISK.SYS

DOS instalará una unidad de disco virtual con el valor por defecto de 64Kb, con sectores de 128 bytes y 64 entradas para archivos.

DEVICE = VDISK.SYS 192 512 12

DOS instalará un disco virtual con sectores de 512 bytes, que soportan 12 archivos de entrada.

Cuando el sistema se vuelve a cargar (reinicializar), *VDISK* mostrará:

```
VDISK Versión 3.2 disco virtual C:  
Entradas en directorio ajustadas  
Tamaño de buffer:192KB  
Tamaño de sector:512  
Entradas en directorio:16
```

DRIVPARM

La versión 3.2 del DOS nos permite modificar las características de los dispositivos con bloques en el momento de arrancar el sistema. El formato de entrada **DRIVPARM** es:

```
DRIVPARM unidad [/checkdoor] [/f:factorforma]  
[/h:numerocabeza] [/noextraíble] [/h:sectores]  
[t:pistas]
```

Donde ocurre lo siguiente:

Unidad es el número de unidad lógica que modifica las características de (A = 0, B = 1, C = 2). El valor debe estar en el intervalo de 0 a 255.

/checkdoor indica que se requiere comprobación de que la puerta de la unidad está cerrada.

Factorforma especifica la forma del dispositivo, según se indica en la Tabla de Factores de forma y características.

Númerocabeza especifica el número de cabeza máximo. El valor debe estar en el intervalo entre 1 y 99.

/noextraíble especifica que el dispositivo no es extraíble (removible).

Sectores especifica el número de sectores por pista.

Pistas indica el número de pistas por cara.

Factor de forma	Características
0	320/360Kb
1	1.2Mb
2	720Kb
3	8 pulgadas, simple densidad
4	8 pulgadas, doble densidad
5	disco rígido
6	unidad de cintas
7	otros dispositivos

Factores de forma y características.

La mayor parte de los usuarios no necesita usar nunca la entrada **DRIVPARM**. Si necesita modificar las características de un dispositivo por bloques, revise antes cuidadosamente el manual de referencia técnico que acompañaba al dispositivo.

FCBS

(Por defecto **FCBS=4.0**). Las primeras versiones del DOS utilizaban bloques de control de archivo (**FCB**) para seguir la pista de cada archivo abierto. Un bloque de control de archivos contiene el estado y estructura de un archivo concreto. Si no se están ejecutando programas de aplicación antiguos, no debería ser necesario preocuparse nunca acerca de la entrada **FCBS** en *CONFIG.SYS*. Por el contrario, si se empiezan a experimentar problemas con los programas más antiguos a medida que se avanza hacia versiones más modernas del DOS, se debe modificar la entrada **FCBS** en *CONFIG.SYS* de esta forma:

FCBS= archivasmáximos, archivosprotegidos

Donde ocurre lo siguiente:

Archivasmáximos es el máximo número de archivos que puede abrir el DOS concurrentemente a través de bloques de control de archivo. El intervalo de valores para archivos es desde 1 hasta 255.

Archivosprotegidos especifica el número de archivos que el DOS no puede cerrar automáticamente cuando necesita abrir archivos nuevos a través de bloques de control de archivos. Este valor debe ser menor o igual que el valor especificado por **archivasmáximos**. El intervalo de valores de protegidos es desde 0 hasta 255.

Como se ha indicado anteriormente, los programas más recientes no utilizan los **FCB**. El ejemplo siguiente permite al DOS abrir 12 archivos simultáneamente a través de **FCB** y evita que el DOS los cierre todos automáticamente:

FCBS = 12,12

FILES

(Por defecto **FILES=8**). El DOS 3.0 introdujo las llaves de archivo para acceder a archivos. La mayoría de las nuevas aplicaciones abren archivos a través de llaves de archivo y a continuación llevan a cabo operaciones de lectura, escritura y cierre, haciendo referencia a la llave del archivo. La entrada **FILES** de **CONFIG.SYS** especifica el máximo número de archivos que puede tener abiertos el DOS simultáneamente. Si sus aplicaciones tienen que abrir más de 8 archivos a la vez, tendrá que modificar la entrada **FILES** de **CONFIG.SYS** así:

FILES=numerarchivos

Donde:

Numerarchivos es el máximo número de archivos que el DOS puede tener abiertos concurrentemente en cualquier momento. El valor debe estar en el rango 8 - 255.

Es importante observar que la entrada de **FILES** especifica el número máximo de archivos que DOS puede abrir en forma concurrente. El DOS sigue limitando cada programa a 20 archivos abiertos. Si no se dedica a programas y si no ejecuta grandes aplicaciones de bases de datos, el valor por defecto de 8 archivos debería ser correcto. En caso contrario, quizá desee permitir 20 archivos abiertos concurrentemente introduciendo:

FILES = 20

Cada llave de archivo por encima del valor por defecto de 8 reserva 48 bytes de memoria. Es importante observar que el DOS proporciona cinco llaves de archivo para cada aplicación que señala a las posiciones que se muestran en la Tabla siguiente. Esto significa que las ocho llaves por defecto del DOS nos permiten en realidad abrir 3 archivos desde dentro de nuestra aplicación.

Llave del archivo	Dispositivo por defecto	Nombre	Operaciones
0	con:	stdin	teclado
1	con:	stdout	pantalla
2	con:	stderr	pantalla
3	aux:	stdaux	dispositivo aux:
4	prn	stdprn	impresora

Llaves de archivos del DOS y aplicaciones

LAST DRIVE

(Por defecto **LASTDRIVE=E**). La entrada **LASTDRIVE** de **CONFIG.SYS** nos permite especificar la identificación de unidad de la última unidad a la que puede referirse el DOS. El formato de la entrada es:

`LASTDRIVE=letra`

Donde:

Letra es la letra (A - Z) de la *última unidad* a la que el DOS puede referirse. Debe ser mayor o igual que el número de unidades instaladas en el sistema.

El ejemplo siguiente establece como última unidad a la unidad C

LASTDRIVE=C

SHELL

La entrada **SHELL** de *CONFIG.SYS* permite especificar un procesador de órdenes alternativo para uso del DOS en lugar de **COMMAND.COM**.

El formato de la entrada es:

SHELL=[unidad:]especificación_de_archivo

Donde:

especificación_de_archivo es el archivo que contiene el programa que debe servir como nuevo procesador de órdenes.

La entrada siguiente define al archivo *MIPROCES.EXE* como procesador de órdenes:

SHELL = MIPROCES.EXE

La mayoría de los usuarios no necesitará nunca especificar un procesador alternativo de órdenes. Los programadores de sistemas deberían leer primero cuidadosamente el manual de referencia técnica del DOS, antes de instalar su propio procesador de órdenes.

El ejemplo siguiente contiene un archivo *CONFIG.SYS* para un IBM PC-AT con 512Kb de memoria:

FILES = 20
BUFFERS = 25
DEVICE = ANSI.SYS
DEVICE = VDISK 192 512 12

*Agradecemos a la División
de Sistemas e Informática
del SENA, Regional
Bogotá y Cundinamarca,
su colaboración para la
edición y producción de
este documento.*

